

Soluções industriais para visualização de dados em tempo-real

Ana Maria Moutinho Pedroso Pereira

Dissertação de Mestrado

Orientador: Prof. António Pessoa de Magalhães



Mestrado Integrado em Engenharia Mecânica

Ramo de Automação

Janeiro de 2018

Aos meus pais, irmã e amigos,

RESUMO

A evolução industrial das últimas décadas abriu as portas aos sistemas de automação e com eles às interfaces Homem-máquina, que discretamente foram evoluindo, transformando-se num elemento fulcral do meio industrial atual. De modo a disponibilizar a informação da fábrica em tempo útil a trabalhadores cuja capacidade de ação e de tomada de decisão é fortemente ligada a essa acessibilidade, como é o caso de engenheiros de manutenção e de supervisão, surgem novas abordagens, como as tecnologias *Web*, que permitem o acesso a dados em qualquer ponto da fábrica com recurso a computadores e/ou a dispositivos móveis.

É neste contexto que surge a presente dissertação, cujo principal objetivo é demonstrar a capacidade de a tecnologia atual dar resposta às necessidades de informação em tempo-real sentidas pelos engenheiros industriais. Desta forma, o documento aqui produzido descreve o desenvolvimento de soluções HMI, apoiadas em diferentes opções tecnológicas, e testadas em cenários industriais virtuais, criados no *software* de emulação *Factory I/O* da *Real Games*.

Numa primeira fase, é realizada uma análise quanto à evolução, desafios, normas, boas práticas e especificações funcionais e operacionais relacionadas com o desenvolvimento de soluções HMI. Daqui segue-se o levantamento de aplicações típicas de interação Homem-máquina na indústria e das necessidades de informação, segmentadas em três níveis de interação: operação, manutenção e supervisão. Posteriormente, é feito um levantamento do estado de arte das tecnologias para visualização industrial, sendo selecionadas duas para implementação no projeto: uma da *3S-Smart Software Solutions* e outra da *Siemens*.

Numa segunda fase, dividida em duas partes, a primeira dedicada à *3S-Smart Software Solutions* e a segunda à *Siemens*, são criados os cenários virtuais e os respetivos modelos de controlo, seguindo-se a especificação e desenvolvimento das interfaces, para cada uma das tecnologias utilizadas. Estas são comparadas entre si, sendo identificados os pontos fortes e fracos de cada uma.

Por fim, os testes realizados a ambas as soluções desenvolvidas permitem validar a utilização de tecnologias *Web*, quer via *browser*, quer via *App*, como uma ferramenta complementar à HMI tradicional, tendo sido possível monitorizar e controlar remotamente os cenários virtuais desenvolvidos no *Factory I/O* com as duas alternativas.

Palavras-Chave: Interfaces Homem-máquina (HMI), Tecnologia *Web*, Acesso remoto e mobilidade, Visualização de dados em tempo-real, Automação industrial, Ambientes virtuais, Cenários industriais.

ABSTRACT

Industrial solutions for real time data visualization

The industrial evolution over the last decades made way for automation systems. Along came the Human-machine interfaces, which have discreetly evolved, becoming a staple of today's industrial environment. For example, a factory's service or supervisor engineer requires real-time data in order to efficiently undertake actions and make decisions. To do that, new approaches emerge, such as Web technologies, which allow data access anywhere on the factory floor by using computers and/or mobile devices.

In this context, the main objective of this dissertation is to demonstrate the ability of current technology to answer the real-time information needs of industrial engineers. Given this, the present document describes the development of HMI solutions, supported by different technologies, which were tested in virtual industrial scenarios created using the emulation software *Factory I/O* from *Real Games*.

At an early stage, a study regarding the evolution, challenges, standards, good practices, functional and operational specifications related with the development of HMI solutions is made. From there, we gather both the industry's standard Human-machine interaction applications, and data needs, segmented in three categories of interaction: operation, service and supervision. Afterwards, two software from the state of the art industrial visualization technologies are selected for implementation in this project: one from *3S-Smart Software Solutions* and other from *Siemens*.

At a later stage, virtual scenarios and their control models are generated, first dedicated to *3S-Smart Software Solutions* and then to *Siemens*, followed by the specification and development of interfaces for each technological platform used. The solutions are compared between them and both strong and weak points discussed.

Finally, the tests to both solutions presented above, allow validation of Web technologies, either via browser or App, as a complementary tool to traditional HMI, as it was possible to remotely control and monitor the virtual scenarios developed in *Factory I/O* using both software alternatives.

Keywords: Human-Machine Interfaces (HMI), Web technology, Remote access and mobility, Real time data visualization, Industrial automation, Virtual environments, Industrial scenarios.

AGRADECIMENTOS

Em primeiro lugar gostaria de agradecer ao meu orientador, Professor António Pessoa de Magalhães, pela oportunidade de desenvolver um trabalho tão desafiante a nível pessoal, assim como por todo o apoio, motivação e conhecimento que sempre me transmitiu. Por tudo isto, o meu sincero obrigada.

Gostaria ainda de agradecer aos meus colegas de curso que de uma forma ou de outra contribuíram para o meu crescimento enquanto pessoa e futura profissional de engenharia; em particular ao Tiago Abreu e ao Silvino Machado, pelas conversas e discussões abertas que em muito me ajudaram para a realização deste trabalho. Também ao Sr. Joaquim Silva, técnico responsável do Laboratório de Óleo-hidráulica, pela disponibilização de um local alternativo de trabalho nos dias de maior calor e pela boa disposição com que sempre me recebeu, um sincero obrigada!

Às pessoas que me acompanharam nestes últimos anos, Ferro, Catarina, Coelho, Mankind, Silva, Hugo e Sérgio, a quem tenho o maior orgulho de chamar amigos, um sentido agradecimento pela vossa amizade e companheirismo, posso afirmar com toda a certeza que este percurso que agora termino não teria sido o mesmo sem vocês!

À minha querida amiga Maria Dias da Costa, irmã que me acompanhou desde o dia em que entrei nesta instituição, agradeço pela motivação e inspiração que sempre me transmitiu, assim como pelo exemplo que marcou, incitando-me sempre a ser a melhor versão de mim própria. Obrigada do coração.

Por fim, e de maior importância, gostaria de agradecer à minha família: aos meus pais, que sempre me apoiaram e me proporcionaram as melhores oportunidades possíveis, transmitindo-me os valores que fazem de mim a pessoa que sou hoje; e à minha irmã, que sempre considereei um exemplo a seguir, agradeço pela barra tão alta, que me obriga diariamente a trabalhar para a conseguir superar.

ÍNDICE DE CONTEÚDOS

RESUMO	V
ABSTRACT	VII
AGRADECIMENTOS.....	IX
ÍNDICE DE CONTEÚDOS	XI
ÍNDICE DE FIGURAS	XV
ÍNDICE DE TABELAS	XIX
ABREVIATURAS, ACRÓNIMOS E SIGLAS.....	XXI
1 INTRODUÇÃO.....	1
1.1 OBJETIVOS DO PROJETO DE DISSERTAÇÃO.....	3
1.2 ESTRUTURA DA DISSERTAÇÃO.....	4
2 INTERAÇÃO HOMEM-MÁQUINA EM AMBIENTES INDUSTRIAIS.....	5
2.1 EVOLUÇÃO E TENDÊNCIAS DAS SOLUÇÕES HMI.....	5
2.1.1 <i>Leque aplicacional das HMIs.....</i>	7
2.1.2 <i>HMI: Desafios a superar.....</i>	8
2.2 ASPETOS DE DESENVOLVIMENTO DE SOLUÇÕES HMI	10
2.2.1 <i>Aspetos funcionais e operacionais.....</i>	10
2.2.2 <i>Conceção de ecrãs.....</i>	11
2.3 NÍVEIS DE INTERAÇÃO	14
2.3.1 <i>Operação.....</i>	15
2.3.2 <i>Manutenção.....</i>	15
2.3.3 <i>Supervisão.....</i>	15
2.4 EXEMPLOS DE APLICAÇÕES INDUSTRIAIS.....	16
2.4.1 <i>Sistemas de transporte tipo conveyor.....</i>	16
2.4.2 <i>Sistema de paletização</i>	17
2.5 TRABALHO PROPOSTO	19
2.6 SÍNTESE.....	19
3 ESTADO DA ARTE DAS TECNOLOGIAS DE VISUALIZAÇÃO INDUSTRIAL.....	21
3.1 SOLUÇÕES DE <i>HARDWARE</i>	21
3.1.1 <i>Painel embutido</i>	21
3.1.2 <i>Painel móvel.....</i>	23
3.2 SOLUÇÕES POR <i>SOFTWARE</i>	24
3.2.1 <i>SoftPLC</i>	25

3.2.2	<i>SoftHMI</i>	26
3.3	MAIS-VALIAS DAS TECNOLOGIAS <i>WEB</i>	27
3.4	SOLUÇÕES TECNOLÓGICAS A EXPLORAR NA DISSERTAÇÃO	30
3.4.1	<i>Solução tecnológica da 3S-Smart Software Solutions GmbH</i>	31
3.4.2	<i>Solução tecnológica da Siemens</i>	32
3.5	SÍNTESE	33
4	DESENVOLVIMENTO DE SOLUÇÕES DE VISUALIZAÇÃO EM CODESYS.....	35
4.1	APLICAÇÃO DE SUPORTE E TESTE	35
4.1.1	<i>Aplicação de paletização</i>	36
4.1.2	<i>Cenários de falha a considerar</i>	39
4.2	ESPECIFICAÇÃO DE INTERFACES.....	40
4.2.1	<i>Especificação da interface de operação</i>	40
4.2.2	<i>Especificação da interface de manutenção</i>	43
4.2.3	<i>Especificação da interface de supervisão</i>	44
4.3	CODESYS	46
4.3.1	<i>Deteção de situações anómalas</i>	46
4.3.2	<i>Visualizações: objetos e funcionalidades</i>	47
4.4	PROGRAMAÇÃO DE INTERFACES.....	49
4.4.1	<i>Interface do operador</i>	51
4.4.2	<i>Interface de manutenção e supervisão</i>	53
4.5	DISCUSSÃO DOS RESULTADOS.....	56
4.6	SÍNTESE	58
5	DESENVOLVIMENTO DE SOLUÇÕES DE VISUALIZAÇÃO EM WINCC	59
5.1	TECNOLOGIA <i>SIEMENS</i>	59
5.1.1	<i>Painel embutido</i>	60
5.1.2	<i>PLC</i>	61
5.1.3	<i>Clientes “App”</i>	62
5.2	APLICAÇÃO DE SUPORTE E TESTE	63
5.2.1	<i>Aplicação de encaminhamento</i>	64
5.2.2	<i>Cenários de falha a considerar</i>	65
5.3	ESPECIFICAÇÃO DE INTERFACES.....	66
5.3.1	<i>Especificação da interface do operador</i>	66
5.3.2	<i>Especificação da interface do engenheiro de manutenção</i>	67
5.3.3	<i>Especificação da interface do engenheiro de supervisão</i>	67

5.3.4	Árvore de ecrãs	68
5.4	TIA PORTAL: STEP 7 E WINCC.....	69
5.4.1	Deteção de situações anómalas	69
5.4.2	Visualizações: objetos e funcionalidades	70
5.5	PROGRAMAÇÃO DAS INTERFACES.....	72
5.5.1	Configuração do servidor Web do S7-1200.....	75
5.5.2	Acesso com a App SIMATIC S7	77
5.6	DISCUSSÃO DOS RESULTADOS	79
5.7	BALANÇO DAS SOLUÇÕES TECNOLÓGICAS IMPLEMENTADAS.....	80
5.8	SÍNTESE.....	82
6	CONCLUSÕES E PERSPETIVAS DE TRABALHO FUTURO	83
6.1	TRABALHOS FUTUROS	85
	REFERÊNCIAS.....	87
A	ANEXO: DETALHES DO DESENVOLVIMENTO DE SOLUÇÕES EM CODESYS.....	91
A.1	MODELOS DE CONTROLO DO PALETIZADOR	91
A.2	CONFIGURAÇÃO DA COMUNICAÇÃO POR “DATA SOURCE”	98
B	ANEXO: DETALHES DO DESENVOLVIMENTO DE SOLUÇÕES EM WINCC.....	101
B.1	MODELOS DE CONTROLO DO ENCAMINHADOR.....	101

ÍNDICE DE FIGURAS

FIGURA 2.1 – COMPARAÇÃO ENTRE INTERFACES DA DÉCADA DE 70, NOMEADAMENTE (A) UM PAINEL DE CONTROLO E (B) UMA PAREDE DE CONTROLO, E INTERFACES ATUAIS, NOMEADAMENTE (C) UMA HMI MODERNA E (D) UMA SALA DE CONTROLO COM SISTEMA SCADA.	6
FIGURA 2.2 – VISUALIZAÇÃO EMBARCADA PARA COMANDO E MONITORIZAÇÃO DE UM EQUIPAMENTO [5].	8
FIGURA 2.3 – VISUALIZAÇÃO GERAL DA FÁBRICA COM RECURSO A VÁRIOS ECRÃS, PARA ESTRUTURAÇÃO DA INFORMAÇÃO DISPONÍVEL PROVENIENTE DE VÍDEO-CÂMARAS, DE MANUAIS GUIA DO UTILIZADOR, DE REGISTO DE TENDÊNCIAS E ATÉ DE TRABALHO COLABORATIVO; BASEADA EM PC [2].	8
FIGURA 2.4 – ACESSIBILIDADE A INFORMAÇÃO NO CHÃO DA FÁBRICA UTILIZANDO SMARTPHONES, TABLETS E PORTÁTEIS [2].	9
FIGURA 2.5 – EXEMPLO DE UMA VISUALIZAÇÃO GRÁFICA INAPROPRIADA PARA CONTROLO OPERACIONAL [17].	12
FIGURA 2.6 – PERCEÇÃO SITUACIONAL UTILIZANDO GRÁFICOS [18].	12
FIGURA 2.7 - MÉTODO DE DESIGN DE INTERFACES CENTRADO NO UTILIZADOR [20].	13
FIGURA 2.8 – EXEMPLO DE UMA INTERFACE GRÁFICA COM 2 NÍVEIS DE HIERARQUIZAÇÃO DE INFORMAÇÃO, CONVENIENTE SINALIZAÇÃO DE ALARMES E UTILIZAÇÃO DE CORES [17].	13
FIGURA 2.9 – EXEMPLO DE UM SISTEMA DE CONVEYORS NUMA LINHA PARA EMBALAMENTO DE CHOCOLATES COM A RESPECTIVA INTERFACE [22].	17
FIGURA 2.10 – PALETIZADOR TOPTier, COM HMI À ESQUERDA [23].	18
FIGURA 2.11 – EXEMPLO DE DUAS VISUALIZAÇÕES DA HMI DO PALETIZADOR TOPTier A NÍVEL DO OPERADOR, EXISTINDO À DIREITA UMA SECÇÃO DEDICADA À MANUTENÇÃO [23].	18
FIGURA 3.1 – PAINÉIS EMBUTIDOS SIEMENS SIMATIC BASIC HMI, 2ª GERAÇÃO [24].	22
FIGURA 3.2 – PAINEL EMBUTIDO OMRON NB5Q COM COMUNICAÇÃO SIMULTÂNEA A PLC E INVERSOR.	22
FIGURA 3.3 – (A) HMI-PLC MOELLER SEM I/O INTEGRADO [28] E (B) PLC+HMI UNITRONICS SAMBA SERIES COM I/O INTEGRADO [29].	23
FIGURA 3.4 – PAINÉIS MÓVEIS DA SIEMENS SIMATIC HMI, (A) VERSÃO WIRED 2ª GERAÇÃO, (B) VERSÃO WIRELESS 1ª GERAÇÃO [31] E (C) PAINEL MÓVEL WIRED KUKA REISPAD LIGADO AO CONTROLADOR KRC ROBOTSTAR [32].	24
FIGURA 3.5 – APLICAÇÃO DE CONTROLO NUM SOFTPLC CODESYS CONTROL WIN V3 COM AS EXTENSÕES TARGETVISU E WEBVISU PARA VISUALIZAÇÃO DE DADOS EM TEMPO-REAL.	25
FIGURA 3.6 – DIFERENTES FORMAS DE VISUALIZAÇÃO DE INFORMAÇÃO COM A TECNOLOGIA DA 3-S SMART SOFTWARE SOLUTIONS GMBH, NOMEADAMENTE CODESYS TARGETVISU, CODESYS WEBVISU E CODESYS HMI [37].	26
FIGURA 3.7 – REPRESENTAÇÃO ESQUEMÁTICA SIMPLES DE UMA REDE INDUSTRIAL.	28

FIGURA 3.8 – ARQUITETURA DA SOLUÇÃO A IMPLEMENTAR COM A TECNOLOGIA CODESYS V3.5.	31
FIGURA 3.9 – ARQUITETURA DA SOLUÇÃO A IMPLEMENTAR COM A TECNOLOGIA DA SIEMENS.	32
FIGURA 4.1 – CÉLULA DE PALETIZAÇÃO DESENVOLVIDA NO FACTORY I/O.	36
FIGURA 4.2 – SUBSISTEMAS DE UM EQUIPAMENTO DE PALETIZAÇÃO.	37
FIGURA 4.3 – DETALHES DO SUBSISTEMA C – ALIMENTAÇÃO DE CAIXAS E FORMAÇÃO DE CAMADA.	38
FIGURA 4.4 – ÁRVORE DE ECRÃS DA INTERFACE DO OPERADOR.	42
FIGURA 4.5 – ÁRVORE DE ECRÃS DA INTERFACE DE MANUTENÇÃO E DE SUPERVISÃO.	45
FIGURA 4.6 – CONFIGURAÇÃO DE GESTÃO DE UTILIZADORES EM CODESYS.	47
FIGURA 4.7 – OBJETOS BÁSICOS DA TOOLBOX DO CODESYS, DE SALIENTAR O RETÂNGULO, A IMAGEM E A FRAME.	47
FIGURA 4.8 – OBJETOS DE CONTROLO DA TOOLBOX DO CODESYS.	48
FIGURA 4.9 – OBJETOS COMPLEXOS DA TOOLBOX DO CODESYS, DE SALIENTAR A TABELA DE ALARMES, O REGISTO DE TENDÊNCIAS E O WEB BROWSER.	48
FIGURA 4.10 – CONFIGURAÇÃO DE ALARMES EM CODESYS.	49
FIGURA 4.11 – ECRÃ DE AUTENTICAÇÃO.	49
FIGURA 4.12 – TEMPLATE PARA A INTERFACE DO OPERADOR.	50
FIGURA 4.13 – TEMPLATE PARA A INTERFACE DE MANUTENÇÃO E DE SUPERVISÃO.	50
FIGURA 4.14 – ECRÃ MENU PRINCIPAL (“HOME SCREEN”) DA INTERFACE DE OPERAÇÃO.	51
FIGURA 4.15 – ECRÃ CONTROLO AUTOMÁTICO DO PALETIZADOR 1 DA INTERFACE DE OPERAÇÃO.	51
FIGURA 4.16 – EXEMPLO DE ECRÃ DE CONTROLO MANUAL DO PALETIZADOR 1 DA INTERFACE DE OPERAÇÃO. .	52
FIGURA 4.17 – ECRÃ DE ALARMES DA INTERFACE DE OPERAÇÃO.	52
FIGURA 4.18 – ECRÃ DE INFORMAÇÃO (ESQUERDA) E ECRÃ DE AJUDA (DIREITA) DA INTERFACE DE OPERAÇÃO.	53
FIGURA 4.19 – ECRÃ DO MODELO COMPORTAMENTAL DO PALETIZADOR 1, DA INTERFACE DE MANUTENÇÃO.	53
FIGURA 4.20 – ECRÃ DE REGISTO DE VARIÁVEIS DA INTERFACE DE MANUTENÇÃO.	54
FIGURA 4.21 – ECRÃ DE MANUTENÇÃO DA INTERFACE DE MANUTENÇÃO.	54
FIGURA 4.22 – ECRÃ DE INFORMAÇÃO (ESQUERDA) E ECRÃ DE AJUDA (DIREITA) DA INTERFACE DE MANUTENÇÃO.	55
FIGURA 4.23 – ECRÃ DE ESTATÍSTICAS DA INTERFACE DE SUPERVISÃO.	55
FIGURA 4.24 – ECRÃ DE EVENTOS DA INTERFACE DE SUPERVISÃO.	55

FIGURA 4.25 – ECRÃ DE INFORMAÇÃO (ESQUERDA) E ECRÃ DE AJUDA (DIREITA) DA INTERFACE DE SUPERVISÃO.	56
FIGURA 4.26 – JANELA DE GESTÃO DE UTILIZADORES, DA INTERFACE DE OPERAÇÃO E DA INTERFACE DE SUPERVISÃO.	56
FIGURA 5.1 – HMI SIMATIC KTP600 BASIC COLOR PN [47].	60
FIGURA 5.2 – PLC SIMATIC S7-1200, CPU 1214C AC/DC/RLY, COM 2 MÓDULOS DE COMUNICAÇÃO SÉRIE.	62
FIGURA 5.3 – ENCAMINHADOR DE CAIXAS, CONSOANTE A SUA DIMENSÃO, DESENVOLVIDO NO FACTORY I/O. .	64
FIGURA 5.4 – ÁRVORE DE ECRÃS DO PAINEL EMBUTIDO HMI SIMATIC KTP600.	68
FIGURA 5.5 – OBJETOS BÁSICOS DO WINCC BASIC V13.	70
FIGURA 5.6 – ELEMENTOS DO WINCC BASIC V13.	70
FIGURA 5.7 – CONTROLOS DO WINCC BASIC V13.	70
FIGURA 5.8 – CONTROLOS DO WINCC BASIC V13, JANELAS POPUP E INDICADOR DE ALARMES.	70
FIGURA 5.9 – CONFIGURAÇÃO DA GESTÃO DE UTILIZADORES EM WINCC BASIC V13.	71
FIGURA 5.10 – CONFIGURAÇÃO DE ALARMES DISCRETOS EM WINCC BASIC V13.	71
FIGURA 5.11 – ECRÃ DE AUTENTICAÇÃO.	72
FIGURA 5.12 – ECRÃ DO MENU PRINCIPAL (“HOME SCREEN”), VISTO POR UM OPERADOR (ESQUERDA), UM ENGENHEIRO DE MANUTENÇÃO (CENTRO) E UM ENGENHEIRO DE SUPERVISÃO (DIREITA).	73
FIGURA 5.13 – ECRÃ DE CONTROLO, COM DIFERENTES MODOS DE OPERAÇÃO SELECIONADOS.	73
FIGURA 5.14 – ECRÃS DE MONITORIZAÇÃO (CENTRO), DE ALTERAÇÃO DE VARIÁVEIS (ESQUERDA) E DE REGISTO DE VARIÁVEIS (DIREITA).	74
FIGURA 5.15 – ECRÃ DE INFORMAÇÃO DO SISTEMA (ESQUERDA), JANELAS POPUP (CENTRO) E ECRÃ DE VISUALIZAÇÃO DE ALARMES (DIREITA).	74
FIGURA 5.16 – ECRÃ DE MANUTENÇÃO – VISTA GERAL (ESQUERDA), ECRÃ DE MANUTENÇÃO – TAPETE ENTRADA (CENTRO) E ECRÃ DE DIAGNÓSTICOS DE SISTEMA (DIREITA).	75
FIGURA 5.17 – ECRÃ DE GESTÃO DE UTILIZADORES (ESQUERDA) E ECRÃ DE ESTATÍSTICAS (DIREITA).	75
FIGURA 5.18 – CONFIGURAÇÃO DO SERVIDOR WEB DO PLC S7-1200.	76
FIGURA 5.19 – CONFIGURAÇÃO DO SERVIDOR WEB E CORRESPONDENTE PROGRAMAÇÃO PARA ATIVAR PÁGINAS DEFINIDAS PELO UTILIZADOR.	77
FIGURA 5.20 – PÁGINA WEB STANDARD COM LISTA DE TAGS INSERIDA E PÁGINA WEB DEFINIDA PELO UTILIZADOR, QUE PERMITE ACEDER A UM FICHEIRO PDF COM O MANUAL DO UTILIZADOR DO PLC.	77

FIGURA 5.21 – CONFIGURAÇÃO E NAVEGAÇÃO DA APP SIMATIC S7 DA SIEMENS.	78
FIGURA 5.22 – SINALIZAÇÃO DE UM UTILIZADOR REMOTO COM DIREITOS DE OPERAÇÃO.	79
FIGURA A1.1 – GRAFCET COMPORTAMENTAL G0 DE ALTO NÍVEL DOS MODOS DE OPERAÇÃO E DE PARAGEM DO PALETIZADOR.	93
FIGURA A1.2 – GRAFCET COMPORTAMENTAL G1 DO ALIMENTADOR DE PALETES.	94
FIGURA A1.3 – GRAFCET COMPORTAMENTAL G2 DO ELEVADOR DE PALETES.	95
FIGURA A1.4 – GRAFCET COMPORTAMENTAL G3 DO ALIMENTADOR DE CAIXAS.	96
FIGURA A1.5 – GRAFCET COMPORTAMENTAL G4 DO DESCARREGADOR DE PALETES.	97
FIGURA A1.6 – DIAGRAMA DE ESTADOS PARA SINCRONIZAÇÃO DO SOFTPLC CODESYS WIN V3 COM O FACTORY I/O.	98
FIGURA A2.1 – CONFIGURAÇÃO DATA SOURCE PARA AS INTERFACES DE MANUTENÇÃO E DE SUPERVISÃO.	99
FIGURA B1.1 – GRAFCET COMPORTAMENTAL G0 DE ALTO NÍVEL DOS MODOS DE OPERAÇÃO E DE PARAGEM DO ENCAMINHADOR.	103
FIGURA B1.2 – GRAFCET COMPORTAMENTAL G1 DO TAPETE DE ROLOS DE ENTRADA.	104
FIGURA B1.3 – GRAFCET COMPORTAMENTAL G2 DA MESA DE TRANSFERÊNCIA.	105
FIGURA B1.4 – GRAFCET COMPORTAMENTAL G3 DE UM TAPETE DE ROLOS DE SAÍDA.	106

ÍNDICE DE TABELAS

TABELA 4.1 – VARIÁVEIS DE INTERFACE DO CONTROLADOR COM O CENÁRIO VIRTUAL: CONTROLO DO PALETIZADOR.	38
TABELA 4.2 – VARIÁVEIS DE INTERFACE DO CONTROLADOR COM O CENÁRIO VIRTUAL: OPERAÇÃO DO PALETIZADOR.	39
TABELA 4.3 – EVENTOS E ALARMES QUE ATIVAM JANELAS POPUP E RESPETIVA POSSIBILIDADE DE AÇÕES, INTERFACE DO OPERADOR.	43
TABELA 4.4 – EVENTOS E ALARMES QUE ATIVAM JANELAS POPUP E RESPETIVA POSSIBILIDADE DE AÇÕES, INTERFACE DE MANUTENÇÃO E DE SUPERVISÃO.	45
TABELA 5.1 – CARATERÍSTICAS FUNCIONAIS DA HMI SIMATIC KTP600 BASIC COLOR PN [46].	61
TABELA 5.2 – FUNCIONALIDADE DA HMI SIMATIC KTP600 SOB CONFIGURAÇÃO NO WINCC [46].	61
TABELA 5.3 – CARATERÍSTICAS FUNCIONAIS DO PLC S7-1200 CPU 1214C AC/DC/RLY [48].	62
TABELA 5.4 – PRINCIPAIS FUNCIONALIDADES DAS APLICAÇÕES SIMATIC S7 [50] E SIMATIC WINCC SM@RTCLIENT [51]	63
TABELA 5.5 – VARIÁVEIS DE INTERFACE DO CONTROLADOR COM O CENÁRIO VIRTUAL: CONTROLO DO ENCAMINHADOR.	65
TABELA 5.6 – VARIÁVEIS DE INTERFACE DO CONTROLADOR COM O CENÁRIO VIRTUAL: OPERAÇÃO DO ENCAMINHADOR.	65
TABELA 5.7 – CONTAGEM DE BITS NOS PLCs SIMATIC S7 E NO WINCC BASIC V13.	72
TABELA 5.8 – AVALIAÇÃO DAS DUAS SOLUÇÕES ENCONTRADAS, A PRIMEIRA EM CODESYS E A SEGUNDA EM WINCC.	82
TABELA A1.1 – DESCRIÇÃO DAS DIFERENTES ETAPAS DO GRAFCET COMPORTAMENTAL DE ALTO NÍVEL DOS MODOS DE OPERAÇÃO E DE PARAGEM DO PALETIZADOR.	92
TABELA B1.1 – DESCRIÇÃO DAS DIFERENTES ETAPAS DO GRAFCET COMPORTAMENTAL DE ALTO NÍVEL DOS MODOS DE OPERAÇÃO E DE PARAGEM DO ENCAMINHADOR.	102

ABREVIATURAS, ACRÓNIMOS E SIGLAS

ASM	Abnormal Situation Management
AWP	Automation Web Programming
CSS	Cascading Style Sheets
DCS	Distributed Control System
FBD	Function Block Diagram
FEUP	Faculdade de Engenharia da Universidade do Porto
GRAFCET	GRAPhe Fonctionnel de Commande Etape Transition
HMI	Human-Machine Interface
HP-HMI	High Performance Human-Machine Interface
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
IEC	International Electrotechnical Commission
IP	Internet Protocol
ISA	International Society of Automation
ISO	International Organization for Standardization
IWLAN	Industrial Wireless Local Area Network
KPI	Key Performance Index
LD	Ladder Diagram
MES	Manufacturing Execution System
OAS	Open Automation Software
OEM	Original Equipment Manufacturer
OPC DA	OLE for Process Control Data Access
OPC UA	OLE for Process Control Unified Architecture
PC	Personal Computer
PLC	Programmable Logic Controller
POU	Program Organization Unit

SCADA	Supervisory Control and Data Acquisition
SFC	Sequential Function Chart
SOA	Service-oriented Architecture
ST	Structured Text
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

1 INTRODUÇÃO

A interface dos controladores com os humanos é tão importante quanto a interface com os sistemas controlados. Por isso, as HMIs (*Human Machine Interfaces*) são, a todos os níveis, recursos fundamentais da moderna automação, dedicando os principais fabricantes de equipamentos de controlo uma considerável atenção à engenharia destes recursos.

A sofisticação das atuais HMIs contrasta em absoluto com a das interfaces de diálogo de outrora. De facto, e não há muitos anos, as interfaces Homem-máquina pouco mais eram do que pequenos conjuntos de interruptores, potenciómetros, sinalizadores e indicadores de ponteiro que, montados num painel, pouco mais permitiam aos operadores do que colocar em marcha, parar e configurar um ou outro parâmetro de uma máquina, ao mesmo tempo que recebiam informações elementares - e por vezes imprecisas - sobre o funcionamento das mesmas. Mas longe parecem ir esses dias, tão rápido e profundo foi o avanço neste domínio. Atualmente, mesmo em máquinas simples, encontram-se e exigem-se interfaces expeditas e completas, onde facilidade de utilização e versatilidade são aspetos fundamentais.

A generalização dos microcomputadores, particularmente nos equipamentos de controlo, incentivou a revolução das HMIs industriais iniciada no último quarto do século passado e consolidada nas décadas seguintes. A era digital que trouxe à cena os PLCs (*Programmable Logic Controllers*), os CNCs e os robôs industriais, acabou por marcar também, e irreversivelmente, a forma como os operadores interagem com as máquinas. Desse modo, termos até há pouco invulgares e estranhos ao mundo industrial – como “menus”, “itens” e “cliques” – são hoje naturais e compreensíveis à generalidade dos mortais.

Mas se facilidade, naturalidade e qualidade da interação parecem ser, pelo menos para os mais leigos, os resultados mais visíveis da evolução tecnológica das HMIs, a verdade é que o principal trunfo destes recursos reside na quantidade e na qualidade da informação que podem atualmente representar. Efetivamente, o elevado número de ecrãs, de variáveis suportadas, de formas de visualização, de seleção de variáveis, de introdução de valores e de comandos possíveis, conferem às atuais HMIs excepcionais capacidades de representação e introdução de dados. E, se aliarmos a tudo isto as características de comunicação em tempo-

real com múltiplos controladores e servidores de dados, bem como a possibilidade de apresentação seletiva de dados em função das competências dos operadores, facilmente se conclui que os aspetos de ergonomia e estética que saltam à vista nas modernas HMIs são, afinal, e de certa forma, aspetos de segundo plano de uma enorme evolução tecnológica.

Porém, nem tudo são rosas nas modernas HMIs, sendo o custo ainda considerável destes recursos, sobretudo no caso das mais sofisticadas e de maiores dimensões, aliado à sua inevitável “imobilidade”, inconvenientes sentidos frequentemente. Por isso, as instalações industriais tendem a não estar equipadas com mais HMIs do que as estritamente necessárias ao seu funcionamento, tendo os operadores e técnicos de manutenção e supervisão de se deslocarem até junto destas sempre que têm necessidade de interagir com uma máquina.

Ainda assim, mesmo nestes aspetos mais desfavoráveis, as HMIs estão a mudar. Novas práticas e tecnologias de interação Homem-máquina têm surgido nos últimos tempos. Concretamente, a utilização de computadores – industriais e comuns – como elementos de interação com máquinas e equipamentos industriais em detrimento de consolas tácteis especificamente concebidas para o efeito, é já uma solução um tanto frequente, sobretudo para responder às necessidades de engenheiros de manutenção e de supervisão. Deste modo, um engenheiro cuja atividade se desenrola por todo o “chão da fábrica” – i.e., por toda a instalação fabril – pode, em qualquer altura, deslocar-se até um computador próximo e, introduzindo as suas credenciais, obter a informação de que necessita ou dar as ordens pretendidas sem necessidade de se deslocar até junto da máquina ou máquinas em causa.

Mas as novas tecnologias da informação permitem ir mais longe. A possibilidade de um engenheiro industrial fazer do seu dispositivo móvel – telemóvel ou *tablet* – a sua “HMI” é uma realidade facilitada pela adoção das tecnologias *Web* e “sem fios” no meio industrial.

As potencialidades oferecidas pelas duas perspetivas acima referidas constituem a motivação da presente dissertação. Propõe-se então a realização de um trabalho de investigação e desenvolvimento de soluções modernas de interação humana com controladores de máquinas e equipamentos industriais. Porém, e não desprezando as interações típicas dos operadores das máquinas, a investigação partirá das necessidades de interação dos engenheiros de manutenção e supervisão, já que são estes a quem mais interessa a possibilidade de interação a partir de terminais remotos e de dispositivos móveis.

A inovação e utilidade do desenvolvimento proposto têm múltiplas vertentes:

Em primeiro lugar, a sua contextualização em cenários industriais absolutamente realistas e de grande interesse prático. Para isso em muito contribuem as facilidades do *software* de emulação de ambientes industriais utilizado. De facto, esse *software*, para além de

permitir recriar ambientes visuais altamente realistas e flexíveis, permite também recriar situações vividas quotidianamente nas fábricas que exigem interações Homem-máquina.

Em segundo, a utilização de uma sofisticada HMI, materializável por *software* ou *hardware*, programável de acordo com normas modernas e associável a diversos *SoftPLCs*, PLCs físicos, e plataformas para controlo embebido. A esta HMI é aliada uma aplicação inovadora de visualização remota lançada poucos dias antes do início deste trabalho. Como vantagem adicional, a HMI oferece serviços *Web* baseados em HTML 5 (*Hyper Text Markup Language*), permitindo o acesso à mesma a partir de dispositivos remotos, fixos ou móveis.

Em terceiro, a comparação das potencialidades das tecnologias acima referidas com as das modernas *Apps*, ou seja, com as daquelas soluções de *software* leves e multiplataforma que se popularizaram nos últimos anos e que se introduziram já no mundo industrial, sobretudo como soluções – ou parte de soluções - de interação Homem-máquina.

1.1 Objetivos do projeto de dissertação

O objetivo fundamental deste trabalho de dissertação é instruir a sua autora no desenvolvimento e avaliação crítica de soluções industriais de interação Homem-máquina. Os cenários considerados procuram refletir necessidades tipicamente encontradas no mundo industrial, mas têm como pano de fundo essencialmente as exigências e interesses dos engenheiros de manutenção e supervisão.

Nesse sentido, e como primeira meta, importa definir e recriar cenários industriais de interesse prático, semelhantes aos vividos quotidianamente em instalações reais de fabricação discreta (*manufacturing*), onde equipamentos e máquinas automáticas interagem em prol de um objetivo produtivo, mas requerem a atenção e a intervenção, periódica ou esporádica, de engenheiros de manutenção e de supervisão. Assim, justificar, elencar e compreender cenários de interesse, definindo “padrões de necessidades e de soluções” é o ponto de partida deste trabalho. O objetivo seguinte é compreender as tecnologias propostas para resolver estes problemas, identificando as suas principais virtudes e lacunas.

O segundo passo é procurar aliar “acesso remoto” e “mobilidade” às tecnologias de interação clássicas. Definir tendências e encontrar propostas inovadoras nesta matéria é o objetivo a atingir.

O desenvolvimento e a experimentação de aplicações alicerçadas nas tecnologias emergentes do levantamento acima referido é o principal objetivo do trabalho. Naturalmente que as restrições temporais e o limite de recursos materiais que sobre ele pesam obrigam à tomada de diversas opções que a seu tempo serão devidamente justificadas.

A organização e a documentação do trabalho realizado é outro objetivo. Para isso, cada caso de estudo desenvolvido é devidamente arquivado e documentado de forma adequada, o que inclui: uma descrição textual do problema, o ambiente virtual desenvolvido, a lista dos equipamentos de controlo e diálogo utilizados e, naturalmente, o *software* desenvolvido e as configurações utilizadas.

1.2 Estrutura da dissertação

A dissertação está organizada em 6 capítulos, servindo o presente para a introduzir.

O capítulo 2, dedicado a aspetos funcionais de HMIs, situa e justifica as linhas de trabalho a desenvolver na dissertação. Concretamente, aborda a interação Homem-máquina no mundo industrial, procurando situá-la a três níveis: operação, manutenção e supervisão. Cenários típicos de interação com máquinas e instalações são apresentados e discutidos.

O capítulo 3 centra-se em questões tecnológicas. Concretamente, revela o estado da arte da tecnologia disponível a cada tipo de utilização. Com isto procura-se perceber até que ponto a generalidade das soluções oferecidas atualmente responde às necessidades dos utilizadores, tanto em termos funcionais como de acessibilidade. Desta análise resulta a escolha justificada das plataformas tecnológicas a utilizar no presente trabalho.

O capítulo 4 desenvolve casos de estudo que têm por base o *software Codesys*, uma linha de *software* para controlo industrial de grande potencial e atualidade, onde o desenvolvimento de HMIs tem um grande destaque. Soluções de acesso remoto, incluindo soluções móveis, são desenvolvidas e ensaiadas neste capítulo, mostrando as potencialidades da utilização de HMIs *Codesys* em benefício dos engenheiros de manutenção e supervisão.

O capítulo 5 tem propósitos semelhantes ao anterior, mas centra-se em soluções de visualização baseadas em tecnologias *Web*, incluindo as modernas *Apps*. A tecnologia explorada é a disponibilizada por um gigante da automação: a *Siemens*; concretamente, os seus PLCs da linha S7-1200. Esta escolha tem a ver com a crescente popularidade destes autómatos que, graças a um reduzido custo aliado a um excelente desempenho, têm conhecido grande popularidade e disseminação no meio industrial.

O capítulo 6 encerra a dissertação. Nele se resumem as principais conclusões, se salientam os ensinamentos mais relevantes e se sugerem formas de lhe dar continuidade.

2 INTERAÇÃO HOMEM-MÁQUINA EM AMBIENTES INDUSTRIAIS

O presente capítulo expõe a evolução das formas e necessidades de interação entre operadores e máquinas industriais ao longo dos tempos, procurando encontrar e justificar as melhores práticas de *design* de interfaces para o mundo industrial. Definem-se três níveis de interação onde a utilização de HMIs é fulcral:

- operação e comando local de máquinas;
- manutenção, local e remota, de máquinas;
- supervisão de instalações e equipamentos.

Também se descrevem alguns exemplos de aplicações industriais típicas onde o uso de HMIs é recorrente, sendo realçadas as características de interação necessárias em cada caso. Por fim, expõe-se e justifica-se a proposta de trabalho a desenvolver nesta dissertação.

2.1 Evolução e tendências das soluções HMI

De uma forma abrangente, uma HMI pode ser definida como um qualquer equipamento físico ou *software* aplicativo que permite a interação (humana) com uma máquina [1]. Esta definição é válida independentemente da aparência da HMI que, efetivamente, pode ser muito variada, como se observa na figura 2.1. Assim, e se no passado as interações Homem-máquina passavam essencialmente por botoneiras, sinalizadores luminosos e/ou sonoros (no caso de um equipamento singular) ou por “paredes de controlo” repletas de mostradores de ponteiro, registos de tendências e caixas de luzes de alarmes (no caso dos processos industriais), os meios de interação usados nos dias de hoje são consideravelmente diferentes [2, 3].

Com os avanços da eletrónica proporcionados na terceira revolução industrial, a introdução da computação e automação em massa nos ambientes fabris permitiu uma revolução na interação entre “Homem e Máquina”. A incorporação de PLCs como base do controlo flexível de diversos equipamentos de uma fábrica trouxe consigo novas necessidades e facilidades de interação. Desse modo, o operador deixou de interagir maioritariamente com

a máquina (o sistema controlado), e passou a interagir também, e sobretudo, com o controlador. Assim nasceram as primeiras HMIs de interface gráfica, cuja popularidade foi crescendo à medida que a utilização destas se tornou mais económica e funcional que a cablagem de componentes eletromecânicos [4]. Outro aspeto a favor destas interfaces, quando aliadas aos controladores, foi a introdução de um “distanciamento físico” entre o operador e a máquina, conferindo assim mais segurança à interação entre ambos. Também a minimização do espaço ocupado e a maior quantidade de informação disponibilizada por este novo tipo de interface contribuíram para a sua proliferação na indústria, uma vez que um simples painel conseguia combinar múltiplos *displays* num só. Isto é, num único ecrã era possível utilizar um menu de navegação para escolher uma das muitas visualizações disponíveis.



Figura 2.1 – Comparação entre interfaces da década de 70, nomeadamente (a) um painel de controlo e (b) uma parede de controlo, e interfaces atuais, nomeadamente (c) uma HMI moderna e (d) uma sala de controlo com sistema SCADA.

A introdução de interfaces gráficas, cuja funcionalidade é programada, trouxe a necessidade de sistemas operativos (SOs) de tempo-real, assim como de *software* de desenvolvimento (SD) de aplicações, para os dispositivos em causa. No início, os OEMs (*original equipment manufacturers*), que se dedicavam ao fabrico de subsistemas para integração em produtos de outras empresas, colocaram no mercado soluções com SOs e SDs proprietários, com ferramentas específicas e de familiaridade, reutilização e incremento de funcionalidade reduzidas; tudo características dos sistemas fechados da altura [4]. Hoje em dia,

verifica-se uma tendência generalizada para sistemas mais abertos, independentes da plataforma de suporte e que facilitam a reutilização de ferramentas nas mais diferentes gerações de *hardware*. Ferramentas avançadas, como objetos gráficos e receitas, assim como linguagens de programação autônomas como C++ e VB.Net representam a evolução do *software* de desenvolvimento de HMIs, sendo que as linguagens referidas permitem uma construção à medida do controlo e funcionalidade do sistema [4].

Ainda assim, a evolução das HMIs esteve longe de se ficar pelo aspeto físico. Efetivamente, a ampla capacidade de comunicação destas interfaces com diferentes marcas e tipos de controladores e PLCs, por vezes simultânea, é uma mais-valia importantíssima introduzida ao longo do tempo. Para isso, em muito ajudaram os protocolos de comunicação normalizados como RS-232, RS-485, *Ethernet* industrial, OPC UA, *Modbus* TCP/IP, entre muitos outros [5].

Atualmente, cada vez mais a escolha de soluções HMI recai sobre três grandes elementos: o seu *hardware*, o seu sistema operativo e o seu *software* de desenvolvimento, sendo estes responsáveis pela generalidade da evolução registada nos últimos anos. A adição de funcionalidade, quer a nível de conectividade com outros equipamentos e bases de dados, quer a nível da sua integração vertical na pirâmide de automação, transformou o termo HMI em algo incapaz de expressar as suas capacidades modernas, sendo agora vulgar o uso do termo “visualização” em referência a estes sistemas [2].

2.1.1 Leque aplicacional das HMIs

Na automação industrial, seja em produção contínua, por partidas (*batch*) ou discreta (*manufacturing*), as HMIs estão omnipresentes, contribuindo tanto para o comando local de máquinas e equipamentos (individuais ou organizados em células de produção), como para a sua manutenção e respetiva supervisão remota. Os níveis de exigência são obviamente diferentes em cada caso e obrigam ou condicionam as escolhas funcionais e tecnológicas, as quais diferem em função da complexidade e heterogeneidade da aplicação, das competências dos utilizadores e da localização das interfaces de diálogo.

As opções de interação dividem-se em duas grandes categorias: HMIs de painéis embutidos, que serão exploradas no próximo capítulo, e HMIs baseadas em PC. A primeira categoria tem mais interesse para uma simples visualização para comando e monitorização local de uma única máquina – como mostra a figura 2.2. A segunda categoria é usual para visualização integrada da empresa ou fábrica, já que disponibiliza a informação específica e relevante a cada utilizador no momento certo – como mostra a figura 2.3 [5].



Figura 2.2 – Visualização embarcada para comando e monitorização de um equipamento [5].



Figura 2.3 – Visualização geral da fábrica com recurso a vários ecrãs, para estruturação da informação disponível proveniente de vídeo-câmaras, de manuais guia do utilizador, de registo de tendências e até de trabalho colaborativo; baseada em PC [2].

2.1.2 HMI: Desafios a superar

Com a quarta revolução industrial em voga, conhecida como “Indústria 4.0”, espera-se que a HMI tome um papel cada vez mais relevante no acesso em tempo-real aos sistemas ciber-físicos resultantes da informatização da indústria [1, 6]. O utilizador destas interfaces, como elemento mais flexível do sistema ciber-físico, tomará cada vez mais responsabilidade na tomada de decisões, assumindo um espectro mais alargado de tarefas, sendo responsável pela monitorização e supervisão de áreas cada vez mais extensas. Por esse motivo, terá que desempenhar as suas funções em movimento, sendo para tal imperativa a existência de ferramentas que o auxiliem nestas tarefas [6]. O projeto “HMI 4.0 – Nutzergenerierte Bedienhilfen” desenvolvido pelo instituto *Fraunhofer IAO* em parceria com empresas como a *Siemens* e *Mercedes*, entre outras, sugere a HMI como uma forma de partilha de

conhecimento no que diz respeito à resolução de situações anormais [7]. Para além deste, outros estudos procuram aplicar uma abordagem SOA (*service-oriented architecture*) utilizando serviços *Web* de forma a obter interfaces adaptativas dinâmicas contextualizadas, isto é, uma interface capaz de alterar as suas características em função da sua envolvente e das necessidades do utilizador [8, 9]. De facto, esta “adaptabilidade” baseada na perceção do contexto, ou seja, quem, onde, em que envolvente, com que equipamento e em que dispositivo de interface, tem elevado interesse para fazer chegar a informação ao sítio, à hora e ao profissional certos e na modalidade mais apropriada (sonora, visual, tátil) [10]. Estes estudos, assim como outros que procuram integrar realidade virtual e realidade aumentada na interação entre Homem e sistema ciber-físico [6], têm vindo a ser testados e validados pela iniciativa tecnológica *SmartFactory^{KL}*, que conta com cerca de 50 membros, entre os quais se encontram empresas como *Siemens*, *B&R*, *Rexroth*, *Festo*, entre muitas outras [11].

Cada vez mais as HMIs suportam capacidades de recolha e processamento de dados mais avançadas e em tempo-real, o que faz com que os utilizadores destas interfaces possam tomar mais decisões e a uma maior velocidade. Com o objetivo de tornar esta interação o mais intuitiva possível, tende-se para um equipamento cuja interação se assemelhará cada vez mais aos equipamentos eletrónicos lúdicos e domésticos de uso generalizado (figura 2.4). Isto deve-se em parte às expectativas da nova geração de trabalhadores, denominada de geração de nativos digitais, que está intrinsecamente acostumada à interação com tecnologia, reduzindo a curva de aprendizagem necessária ao desempenho das suas funções profissionais na indústria [1, 2].



Figura 2.4 – Acessibilidade a informação no chão da fábrica utilizando smartphones, tablets e portáteis [2].

O próximo passo é então transformar os próprios dispositivos móveis de cada indivíduo, seja um *smartphone*, um *tablet* ou até mesmo um *smart watch*, em equipamentos capazes de interagir com o equipamento industrial e de certa forma complementar à HMI tradicional, conferindo-lhe a mobilidade e a acessibilidade em tempo-real necessária [1, 6, 12, 13]. Para isto em muito contribuem as tecnologias *Web*, que serão abordadas no capítulo 3.

2.2 Aspectos de desenvolvimento de soluções HMI

O desenvolvimento de uma solução HMI é deveras complexo e minucioso, tanto em termos de tempo como de recursos, principalmente se o desenvolvimento não for corretamente especificado e planeado [14]. Um projeto terá sempre que partir da especificação das necessidades funcionais a cobrir, isto é, que funções têm que ser desempenhadas; seguindo-se posteriormente as especificações tecnológicas e operacionais, ou seja, como é que as funções serão implementadas. Ao longo de todos os processos de engenharia, desde o *design*, o *layout* do painel, produção e testes ao controlo de qualidade, é imperativa a consulta de todas as normas e “guias de boas práticas” aplicáveis ao sistema a desenvolver, de modo a garantir a conformidade do projeto com as mesmas [14]. Estas podem ser normas industriais, de segurança (“Diretiva 2006/42/CE” [15]), ergonómicas (“ISO 9241 *Ergonomic requirements for office work with display terminals*” [16] e “ISO 11064 *Ergonomic design of control centres*” [17]), ou orientações gerais sobre a conceção de HMIs (“ISA 101 *Human Machine Interfaces for Process Automation Systems*” [18] e “ASM Consortium *Guidelines: Effective Operator Display Design*” [19]) [20].

No caso particular do desenvolvimento de HMIs, importa referir alguns aspetos gerais do mesmo, sendo dado, posteriormente, algum relevo ao *design* gráfico de visualizações, uma vez que esta etapa terá um impacto tão importante na qualidade da solução final como as etapas iniciais.

2.2.1 Aspetos funcionais e operacionais

A especificação funcional e operacional de uma solução HMI é suposta ter em conta vários fatores, tais como [14, 21]:

- **ambiente operacional** – a exposição a humidade, temperaturas extremas, ambientes hostis e o simples desgaste do equipamento afetam a durabilidade da solução HMI, que se espera que seja pelo menos igual ao ciclo de vida do equipamento a que estará associada.
- **tipo de utilizador** – se para um utilizador passivo a interface deverá ser o mais intuitiva possível, tomando partido da repetibilidade e do emprego de comandos simples, para um utilizador mais avançado poderá ser mais vantajosa a utilização de ferramentas de controlo mais sofisticadas, com diferentes níveis de interação com o equipamento em causa.

- **segurança** – tendo grande parte das falhas nos equipamentos origem em erros humanos, a incorporação de equipamento de segurança adequado nas soluções HMI torna-se obrigatória e de extrema importância.
- **necessidades de comunicação com outros equipamentos** – tanto horizontalmente com equipamentos no chão da fábrica como verticalmente com o MES ou sistemas de logística e inventário; estas comunicações podem ser *hard-wired*, principalmente para sistemas *legacy*, ou então digitais, por barramento série (*serial bus*), por *ethernet*, ou ainda por *wireless*, para os sistemas mais recentes. As comunicações digitais por barramento série são mais vantajosas na medida em que são mais compactas e flexíveis, assim como mais fiáveis e rápidas na entrega de dados em tempo-real.
- **tipo de *feedback*** – sendo este essencial para que haja confirmação da execução de qualquer ação do utilizador da interface; o *feedback* pode ser visual, sonoro, tátil ou uma combinação destes.
- **complexidade da interação (*inputs/outputs*)** – que ditará a tecnologia que implementará determinada função.

Daqui resulta que a escolha tecnológica a nível de *display*, de capacidade interativa de *touchscreen*, de interruptores, de teclas, de controlo de cursor, entre outros, terá que ser adaptada aos fatores acima referidos.

2.2.2 Conceção de ecrãs

Durante muito tempo, as interfaces amplamente implementadas na indústria não estavam otimizadas, tendo falhas a nível de eficiência, eficácia e resposta às necessidades do utilizador, prejudicando o desempenho do último. Isto deve-se em parte às restrições temporais e orçamentais impostas ao desenvolvimento destas interfaces, como também ao pouco conhecimento existente na altura da transição para interfaces gráficas sobre o desenvolvimento das mesmas. Aliando a isto o uso indiscriminado das funcionalidades proporcionadas pelos modernos *softwares* de desenvolvimento, daí resultaram interfaces de usabilidade reduzida, como se pode observar na figura 2.5.



Figura 2.5 – Exemplo de uma visualização gráfica inapropriada para controlo operacional [22].

Hoje em dia, ao combinar as melhores práticas de *design* de interfaces com estudos na área da psicologia cognitiva, surge a metodologia para o desenvolvimento de HP-HMI (*High Performance HMI*). Esta assenta em princípios como a clareza, a uniformidade e o *feedback* da interface, de modo a disponibilizar ao utilizador dados devidamente contextualizados e representados, que lhe permitam uma melhor perceção situacional (*situational awareness*) do processo ou tarefa. Entenda-se por perceção situacional, a perceção de informação relevante, compreensão dessa informação no contexto atual da tarefa e dos objetivos pretendidos, seguida da projeção da evolução da situação, para que as ações necessárias possam ser implementadas [3, 23, 24]. A figura 2.6 exemplifica como a utilização de gráficos favorece uma melhor perceção situacional.

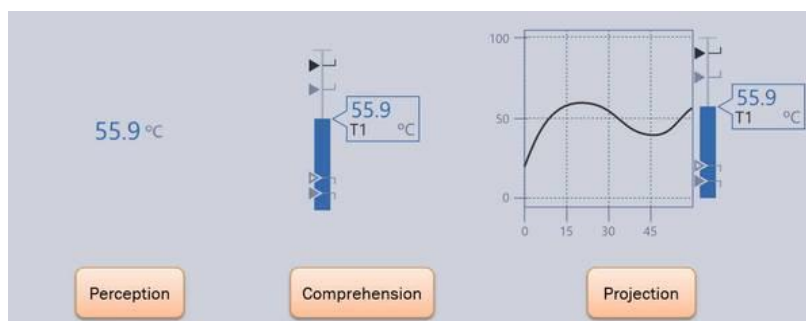


Figura 2.6 – Perceção situacional utilizando gráficos [23].

Para o desenvolvimento de uma interface Homem-máquina é imprescindível o conhecimento das capacidades e limitações de processamento de informação de ambos os intervenientes. Negligenciar este aspeto aumentará a propensão para comandos e

interpretações errôneas, que consoante a aplicação, poderão ter consequências graves. Por essa razão, sugere-se uma metodologia de “*design* centrado no utilizador”, descrita em mais pormenor na figura 2.7, na qual o utilizador alvo da interface tem um papel ativo na fase de desenvolvimento e teste da mesma [21, 25]. Deste método resulta uma interface de elevada usabilidade, com melhor desempenho que interfaces tradicionais [26].

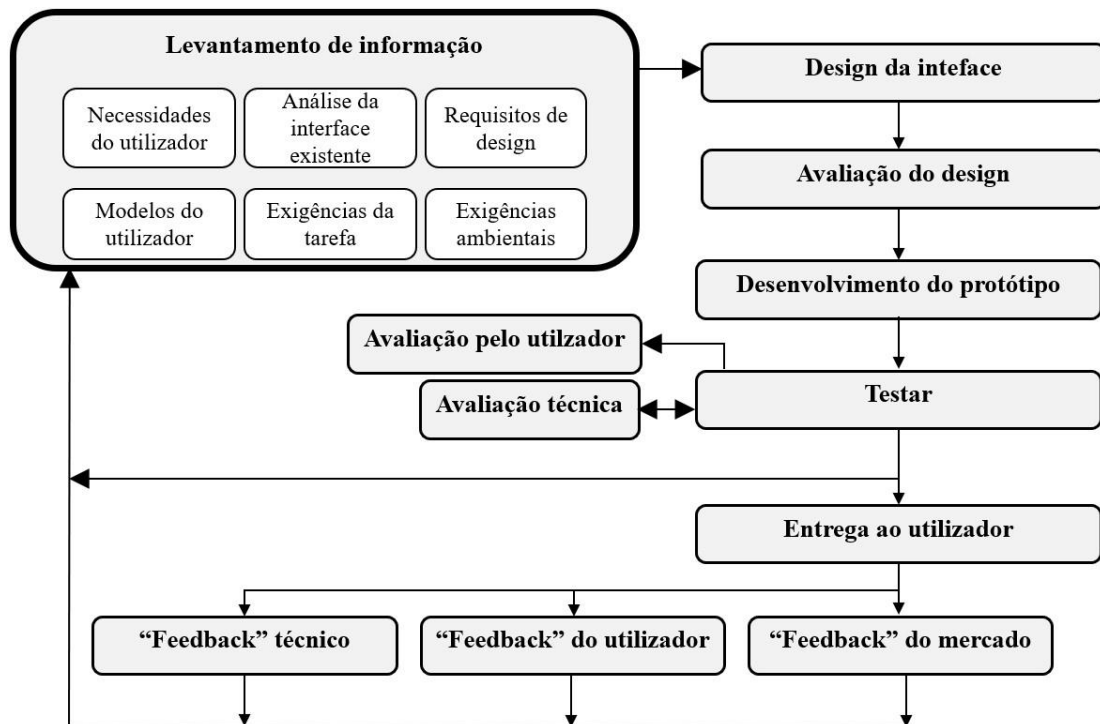


Figura 2.7 - Método de design de interfaces centrado no utilizador [25].

Para atingir a referida perceção situacional, as interfaces desenvolvidas devem implementar a otimização do uso de cores, a conveniente sinalização de alarmes, o desenho minimalista de objetos, a extensa utilização de gráficos, a hierarquização da informação disponibilizada e a utilização eficiente do espaço disponível [3, 23, 24]. A figura 2.8 exemplifica a aplicação dos requisitos acima referidos.

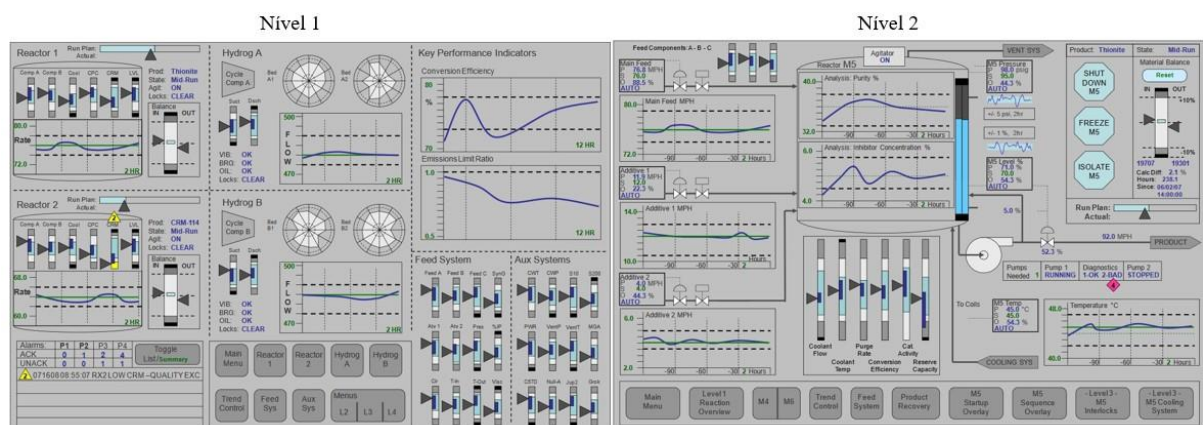


Figura 2.8 – Exemplo de uma interface gráfica com 2 níveis de hierarquização de informação, conveniente sinalização de alarmes e utilização de cores [22].

As características referidas são facilmente justificadas pela forma como o ser-humano processa e armazena a informação visual, com recurso à memória sensorial, de curto e de longo prazo. É na memória de curto prazo, de cariz temporário, que informação de maior destaque é armazenada, até ser provada relevante, e passada à memória de longo prazo. Desta forma, a sobrecarga de fluxo de informação na memória de curto prazo conduzirá a uma possível perda de informação fulcral à resolução de um problema [21]. A figura 2.5 constitui um exemplo de uma HMI que constantemente monopoliza a atenção do utilizador para situações que não requerem excessiva atenção. Por essa razão é preferível utilizar uma representação simplificada dos componentes físicos, assim como cores suaves para o fundo de visualização, sendo as cores mais intensas e apelativas reservadas para indicar situações críticas e estados relevantes ao utilizador. Ainda assim, a sinalização de situações críticas ou de alarme nunca deverá depender somente da cor utilizada, uma vez que o utilizador pode ter restrições quanto à sua perceção de cor (por exemplo daltonismo). Por isso, e para garantir uma acessibilidade universal à informação, a forma da sinalização do alarme deverá desempenhar também um papel de realce e distinção [23, 24].

Esta abordagem ao design de HMIs tem a seu favor muito mais do que o aspeto estético da interface, uma vez que, quando se comparam HMIs de elevado desempenho com as HMIs tradicionais, se verifica que as primeiras favorecem uma melhor performance na deteção e resolução de situações anormais por parte do utilizador [3]. Ou seja, a prevenção de falhas, incluindo a redução de erros operacionais; o aumento da eficiência operacional, melhorando a qualidade da informação, facilitando a aprendizagem e requerendo menor esforço de processamento cognitivo por parte do utilizador são algumas das vantagens mais relevantes deste tipo de interfaces [23, 24].

2.3 Níveis de interação

No quotidiano de uma instalação fabril existe uma constante necessidade de interação e troca de informação entre máquinas ou sistemas e diferentes trabalhadores. E como seria de esperar, nem sempre a informação trocada é a mesma, nem essa troca ocorre com a mesma frequência, nem pelas mesmas razões. Isto é, o tipo de informação que um gestor de fábrica requer de um dado sistema não será certamente a mesma que a informação procurada pelo operador desse mesmo sistema. Da mesma forma, a frequência de interação de um operador que todos os dias interage com uma máquina difere certamente da frequência de interação de um engenheiro de manutenção e essa mesma máquina.

Por esse motivo, e tendo em conta as diferentes necessidades de informação de cada profissional, podem-se definir diferentes níveis de interação Homem-máquina no meio

industrial. Para o trabalho em questão optou-se por abordar apenas três: nível de operação, de manutenção e de supervisão. De notar que a melhor forma de aferir as necessidades de cada um destes intervenientes seria realizando casos de estudo em ambientes industriais reais. No entanto, não podendo contar com essa possibilidade em tempo útil, optou-se por uma revisão na literatura especializada no tema.

2.3.1 Operação

Este nível caracteriza-se pelo contacto permanente ou, pelo menos, bastante próximo, entre o operador e o equipamento. Em termos da informação requerida, este será o nível mais baixo de processamento de dados e o mais básico. O estado de funcionamento do equipamento é uma clara prioridade, assim como avisos e erros do mesmo. Outro ponto de interesse será saber a qualidade da produção, como também as encomendas/ordens de trabalho feitas, no caso da fabricação discreta [8].

2.3.2 Manutenção

Contrariamente ao operador, um engenheiro/técnico de manutenção interage com uma pluralidade de equipamentos, ainda que a interação com cada um deles se dê com menos frequência. Embora a este profissional interessem, tal como ao operador, o estado, os avisos e os erros dos diferentes equipamentos, outras informações são tão ou mais importantes. Concretamente, a informação relativa aos trabalhos de manutenção agendados (manutenção preventiva), aos fornecedores de equipamento, às peças sobressalentes existentes, assim como informação técnica (manuais, diagramas, vídeos, entre outros) desempenha um papel fulcral para o dia-a-dia deste profissional [8]. Para além de tudo isto, a disponibilização de informação de diagnóstico do equipamento seria uma mais-valia, para tornar um trabalho de manutenção excecional mais expedito.

2.3.3 Supervisão

Por fim, o nível de supervisão poderá ser caracterizado como uma extensão do nível do operador. Isto é, para além da informação disponibilizada ao operador, ao supervisor será de particular interesse o acesso a dados que quantifiquem o desempenho da fábrica ou setor, por exemplo KPIs (*key performance indicators*), assim como a informação dos *stocks* existentes e o pessoal disponível [8]. De um modo geral, este profissional é responsável por supervisionar o desempenho de grande parte do equipamento que se encontra no chão da fábrica, sendo dotado das competências necessárias para intervir na resolução de problemas de operação, quando o operador não tem competência para resolver os mesmos. Trata-se assim do nível

elevado de processamento de dados no interior da fábrica, tipicamente auxiliados ou desempenhados por sistemas SCADA (*supervisory control and data acquisition*).

2.4 Exemplos de aplicações industriais

Existe uma infinidade de aplicações industriais envolvendo HMIs, cada uma com necessidades de interfaces específicas. Tentando fazer um breve apanhado das mais relevantes, optou-se por exemplos de fabricação discreta, utilizados tanto na fase de produção como no manuseamento das matérias-primas, componentes e produtos finais.

Apesar das diferenças funcionais inerentes a cada aplicação, é possível delinear um padrão base com os componentes e *layouts* universalmente utilizados. Em termos de componentes físicos: um ecrã tátil é a opção generalizada como interface principal, sendo a sua dimensão variável para satisfazer as necessidades específicas do sistema; a utilização de uma botoneira física de emergência é comum e obrigatória, tendo em conta as diretivas de segurança em vigor; no caso de painéis moveis ou consolas, encontrar um *dead man switch* será também uma forte possibilidade, tipicamente em painéis de interface com robôs; no caso de uma interface gráfica não tátil, a utilização de periféricos como “teclados” ou “ratos” são uma opção. Focando a atenção no layout de uma interface gráfica: um menu para navegação da interface gráfica, assim como dados como data e hora e informação do utilizador são recorrentes.

Testemunhando o padrão acima referido analisam-se em seguida algumas interfaces usadas em aplicações genéricas muito comuns em fabricação, embora com detalhes que variam de caso para caso: os sistemas de movimentação de matérias/produtos e os sistemas de paletização. A utilização deste tipo de sistemas como modelos para o estudo a realizar ao longo desta dissertação permitirá comprovar a vasta aplicabilidade dos resultados obtidos nos mais diferenciados ramos da indústria como se mostra nos pontos que se seguem.

2.4.1 *Sistemas de transporte tipo conveyor*

O sistema de transporte em tapete, como exemplificado na figura 2.9, encontra-se amplamente disseminado na indústria, sendo utilizado para levar um produto X de um ponto A para um ponto B numa fábrica. A junção deste tipo de mecanismo com a funcionalidade de encaminhamento/seleção por cor, tamanho, massa, entre outros, é também comum.



Figura 2.9 – Exemplo de um sistema de conveyors numa linha para embalagem de chocolates com a respetiva interface [27].

Para a execução destas funções, a interface deve permitir aceder e editar a configuração do equipamento. Isto é, em termos de caminhos a realizar quando em modo automático ou quando em modo manual, assim como em outros modos de funcionamento. Para isso serão necessários campos de *inputs*, quer botões quer campos de texto, para que a interação entre o utilizador e o controlador seja possível. Deve também dispor ao operador, meios para controlar a quantidade de matéria/produto transportada.

Do ponto de vista do engenheiro de manutenção deve apresentar as ferramentas de diagnóstico necessárias para a resolução de problemas de funcionamento, seja o estado de um determinado sensor ou atuador, ou o total número de horas de funcionamento de dado componente ou até mesmo históricos de falhas semelhantes já ocorridas ou de ações de manutenção preventiva já realizadas.

No caso da supervisão importa saber os produtos em trânsito, em tipo e quantidade, assim como falhas cuja dimensão afete o funcionamento de equipamentos adjacentes e inviabilize a produção do setor. Para além destes, KPIs da produção, informação quanto à eficiência energética e acesso ao histórico de eventos realizados são também elementos de interesse.

Dada a maior área de trabalho que estes sistemas ocupam, podendo esta desenrolar-se ao longo da fábrica inteira, justificar-se-ia o emprego de um painel móvel para um nível de operação no chão da fábrica, ou então, um sistema centralizado de controlo e supervisão de todo o sistema, isto para desempenhar as funcionalidades de nível mais elevado.

2.4.2 Sistema de paletização

Típico equipamento para *handling* de produtos no final de linha, antes de armazenamento. Dada a flexibilidade e versatilidade das fábricas modernas, as caixas de

embalamento podem ter diferentes dimensões, sendo necessária a mesma flexibilidade para o processo de paletização automático dessas mesmas caixas.

A figura 2.10 representa um paletizador TopTier com um painel de interface gráfico embutido à esquerda.



Figura 2.10 – Paletizador TopTier, com HMI à esquerda [28].

As visualizações disponibilizadas devem permitir todas as operações do equipamento, quer a nível manual, quer a nível automático. Entre as funções necessárias encontram-se a escolha de padrões de paletização predefinidos assim como a adição de novos padrões ao equipamento. Também procedimentos de recuperação e de esvaziamento do equipamento devem ser considerados. Para além destes, a monitorização do sistema utilizando gestão de alarmes, registos históricos do funcionamento da máquina, sistemas de segurança e observando a sequência de operações do equipamento [28]. A figura 2.11 apresenta as duas principais visualizações do paletizador acima referenciado, onde se obtém uma visão geral sobre a funcionalidade pretendida para esta interface.

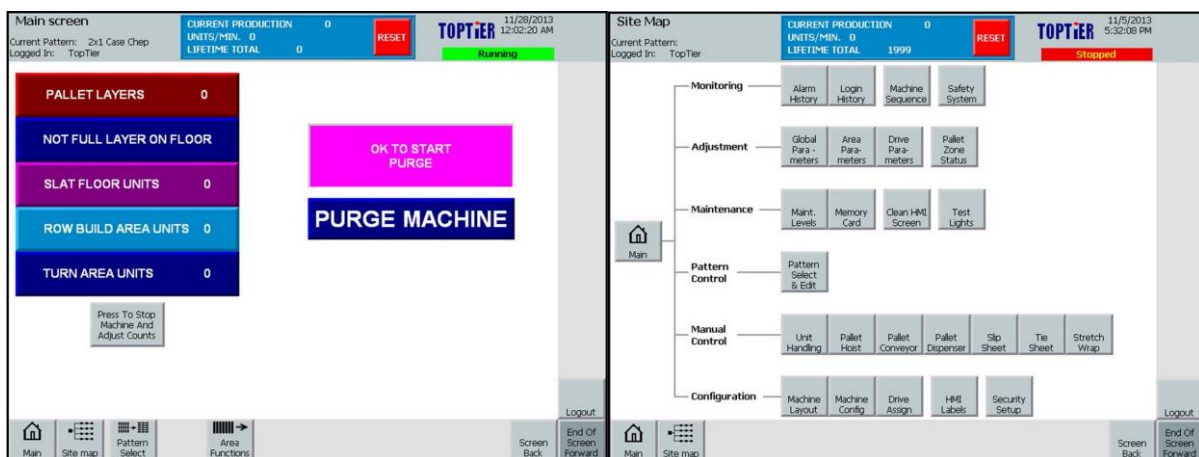


Figura 2.11 – Exemplo de duas visualizações da HMI do paletizador TopTier a nível do operador, existindo à direita uma secção dedicada à manutenção [28].

2.5 Trabalho proposto

Com esta dissertação procurar-se-á desenvolver interfaces para diferentes cenários industriais comuns nos dias de hoje, sendo particularmente estudadas e implementadas as funcionalidades mais vantajosas para melhorar e facilitar o trabalho dos engenheiros das áreas de manutenção e supervisão.

Numa fase inicial serão criados os sistemas industriais de interesse num simulador 3D de ambientes industriais denominado *Factory I/O*.

Seguir-se-á o desenvolvimento tanto do controlo, como de um conjunto de visualizações para cada um dos cenários criados, tendo em conta as necessidades de cada um dos intervenientes: operador, engenheiro de manutenção e engenheiro de supervisão. Numa fase final serão exploradas as vantagens e desvantagens proporcionadas no domínio da mobilidade e acessibilidade pelas modernas *Apps*, face à utilização de serviços *Web*.

De modo a identificar as tecnologias de visualização industrial mais promissoras para a implementação do trabalho proposto, será realizada uma análise dessas mesmas tecnologias no próximo capítulo.

2.6 Síntese

O capítulo que aqui termina evidenciou o quanto evoluiu a interação Homem-máquina na indústria nas últimas décadas, tendo a eletrónica “tomado de assalto” os meios industriais, dando mais funcionalidades às interfaces de outrora. Este capítulo apresentou as interfaces gráficas de hoje como sistemas abertos, independentes da plataforma de implementação, com flexibilidade para suportar diferentes protocolos de comunicação, e onde o *software* de desenvolvimento toma um papel de grande importância no que diz respeito à adição de novas funcionalidades.

Para além disto, mostrou ainda como as interfaces atuais podem assumir novos desafios como parte integrante da fábrica do futuro, assim como salientou a viabilidade de equipamentos como *smartphones* e *tablets* como “janelas” para o controlo e monitorização de equipamentos industriais em tempo-real. Também mostrou vários aspetos funcionais e operacionais, a ter em conta aquando do projeto de uma HMI, tendo sido realçada a importância das normas de conceção gráfica para o elevado desempenho da solução desenvolvida. Por fim, não só listou as necessidades de informação para cada profissional das áreas de operação, manutenção e supervisão, como apresentou dois exemplos de interfaces de aplicações industriais recorrentes.

Em suma, e embora as HMIs sejam muitas vezes encaradas como algo supérfluo ou acessório, sendo alvo de pouco destaque face a equipamentos como os PLCs, comprovou-se que estas são recursos essenciais e de elevada exigência na automação industrial.

3 ESTADO DA ARTE DAS TECNOLOGIAS DE VISUALIZAÇÃO INDUSTRIAL

A oferta de equipamento para automação industrial é cada vez mais vasta e diversificada no que diz respeito ao controlo e à monitorização de ambientes fabris. Por esse motivo, o presente capítulo analisa o estado da arte das tecnologias de visualização industrial, dando relevo às opções mais adequadas e promissoras para a visualização de dados e informações industriais de tempo-real. Para além disso, analisa a tecnologia emergente para acesso remoto ao sistema de controlo e monitorização do equipamento presente no ambiente fabril. Relativamente ao primeiro ponto, serão abordadas as diferentes soluções de *hardware* (painéis e PLCs) e por *software* (*SoftHMI* e *SoftPLC*). Já o segundo ponto focará as tecnologias *Web* e as *Apps*. Daí será delineada e justificada a estrutura tecnológica a explorar nesta dissertação.

3.1 Soluções de *Hardware*

Dada a diversidade de fornecedores de tecnologia de visualização industrial no mercado, serão apenas focados neste capítulo os exemplos representativos do leque aplicacional existente. Serão salientadas as suas funcionalidades mais importantes, assim como a capacidade de munir os seus utilizadores (operador, engenheiro de manutenção ou supervisor) com as ferramentas mais adequadas ao seu trabalho, principalmente no que diz respeito à acessibilidade a informação.

3.1.1 *Painel embutido*

Entenda-se por painel embutido qualquer dispositivo de visualização fisicamente incorporado na máquina ou no seu armário de controlo. Mediante a complexidade da aplicação, existe uma panóplia de opções no que diz respeito ao tamanho de ecrã, processador, memória interna e protocolos de comunicação. Na categoria de painel embutido distinguem-se os dispositivos com protocolo de comunicação proprietário, otimizados para utilização conjunta com equipamento da mesma “marca”; os dispositivos com protocolos de comunicação abertos/normalizados, capazes de serem integrados com uma diversidade elevada de equipamentos de “marcas” diferentes; e por fim os dispositivos com PLC

integrado, designado HMI-PLC, com intuito de aglomerar num único equipamento as funcionalidades de controlo e de monitorização.

Um exemplo de painéis embutidos com protocolo de comunicação proprietário é a gama de painéis SIMATIC da *Siemens* (figura 3.1). Sendo esta última a detentora de uma das maiores cotas de mercado da automação industrial, os seus painéis são concebidos para uma fácil integração com PLCs da *Siemens*, facilitando o desenvolvimento conjunto de aplicações de controlo e de visualização [29].



Figura 3.1 – Painéis embutidos Siemens SIMATIC Basic HMI, 2ª geração [29].

Por outro lado, os dispositivos com protocolos de comunicação abertos/normalizados, como *Modbus* TCP/IP ou OPC UA, são ideais para ambientes industriais com elevada heterogeneidade de equipamentos, não estando limitados à integração com PLCs de um único fornecedor. Como exemplo destaca-se os painéis embutidos da *Omron* (figura 3.2), com elevada versatilidade de comunicação (portas série, *Ethernet* e USB) e compatibilidade com vários PLCs de fornecedores diferentes [30]. As HMIs da série NB da *Omron* permitem a utilização de duas portas em simultâneo, permitindo assim a comunicação com PLCs, inversores, variadores de frequência, controladores de temperatura, leitores de códigos de barras, entre outros [31].



Figura 3.2 – Painel embutido Omron NB5Q com comunicação simultânea a PLC e inversor.

Por fim, a HMI-PLC trata-se de uma solução compacta e económica que combina o processamento de um PLC com a monitorização e controlo de uma HMI [32]. Surge como resposta à redundância introduzida pela sobreposição de funcionalidade entre PLC e HMI, nomeadamente, captura e troca de dados históricos, notificações de alarmes, gestão e segurança, em parte motivada pela proliferação de redes *fieldbus* e da *Ethernet* industrial [32]. A preferência por redes de campo em detrimento de I/Os conduz a dispositivos como os da *Eaton/Moeller* (figura 3.3a), ainda que existam soluções que mantêm I/O integrado na HMI-PLC (figura 3.3b).

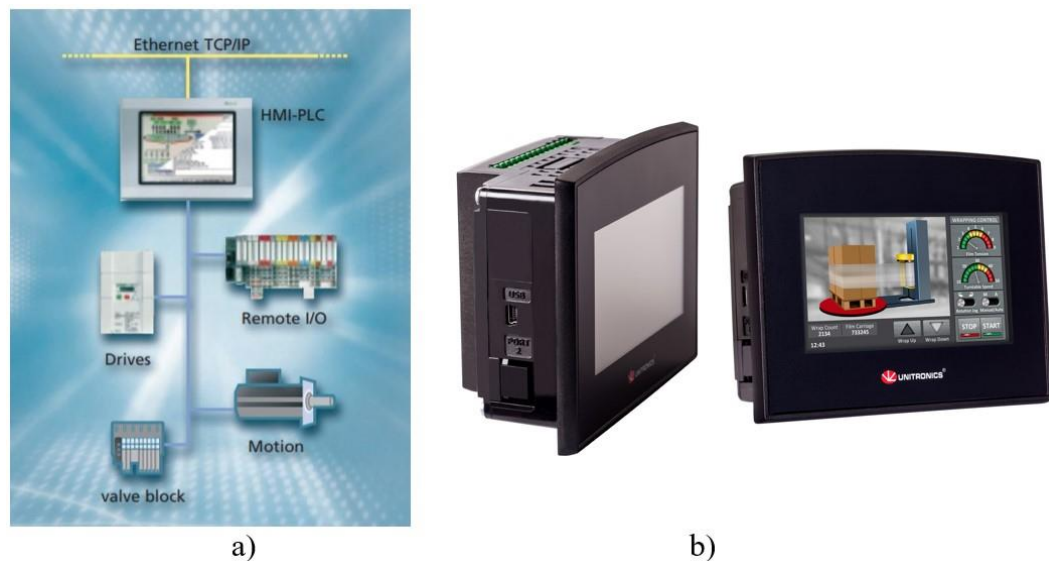


Figura 3.3 – (a) HMI-PLC Moeller sem I/O integrado [33] e (b) PLC+HMI Unitronics Samba Series com I/O integrado [34].

A figura 3.3b representa uma solução PLC+HMI da *Unitronics*, uma empresa exclusivamente dedicada a soluções de *hardware* integradas de PLCs com HMIs. Salienta-se a sua capacidade de comunicação com dispositivos externos: *Ethernet* TCP/IP, envio de e-mails e de SMS, *Modem GSM* (*Global System for Mobile Communications*), *Modbus*, *CANopen*, comunicação série, entre outros [34].

Estes equipamentos constituem uma opção ótima para aplicações de baixa complexidade, como para equipamentos de pequena dimensão, equipamento de embalagem, compressores ou bombas, sistemas de processamento, entre outros [35].

3.1.2 Painel móvel

Um painel móvel enquadra-se face às opções já referidas como o passo inicial da indústria em direção à mobilidade de informação a nível fabril. Trata-se de uma forma de monitorização e controlo prática e vastamente implementada em alguns setores do meio industrial, principalmente na sua vertente *wired* (figura 3.4a), sendo que dos grandes

fabricantes deste tipo de equipamento, apenas a *Siemens* comercializa uma gama de painéis *wireless* (figura 3.4b) [36].

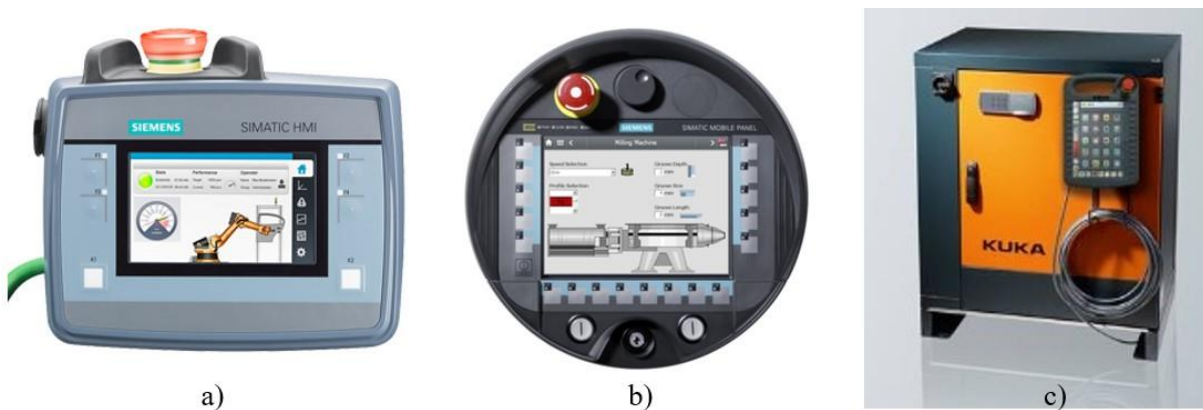


Figura 3.4 – Painéis móveis da Siemens SIMATIC HMI, (a) versão wired 2ª geração, (b) versão wireless 1ª geração [36] e (c) painel móvel wired KUKA reisPAD ligado ao controlador KRC ROBOTstar [37].

Uma aplicação típica de painéis móveis em vertente *wired* é o comando e monitorização de robôs industriais (figura 3.4c) [37]. Neste tipo de aplicação a mobilidade do operador é chave, dada a infinidade de movimentos de um robô industrial pode executar. Já o painel móvel *wireless* traz vantagens para aplicações onde a atividade industrial se desenrola numa área geográfica extensa, como se verifica pela sua utilização na *Heinz Schnorpfeil Baustoff GmbH & Co* [38]. Nesta empresa, o operador pode-se deslocar por toda a instalação e utilizar o painel móvel *wireless* para interagir com todos os equipamentos, o que só é possível garantindo uma total cobertura por uma rede IWLAN.

Se por um lado estes equipamentos se destacam pela sua mobilidade, elevada resistência a choques, humidade, vibração e sujidade, permitindo ao seu utilizador uma maior liberdade na supervisão e controlo do trabalho realizado pela máquina [36], melhorando assim a produtividade da fábrica; por outro lado, o custo e as necessidades adicionais de equipamento de segurança quando se utiliza um painel *wireless* [39], podem suscitar alguma reticência na adoção deste tipo de tecnologia. Para além disso, também a preocupação em garantir que as máquinas possam ser comandadas remotamente de forma segura influencia as decisões atualmente tomadas quanto à implementação de controlo remoto nas diferentes indústrias.

3.2 Soluções por Software

Para além das opções referidas, existem soluções de *software* que podem ser utilizadas em PCs convencionais e industriais, de modo a que estes desempenhem funções tradicionalmente confiadas a equipamentos industriais propositadamente concebidos para o efeito (PLCs e HMIs). Estas designam-se por *SoftPLCs* ou *SoftHMIs*.

3.2.1 SoftPLC

Nesta categoria toma-se como exemplo os *SoftPLC* da *3S-Smart Software Solutions*, o *Codesys Control Win* e o *Codesys Control RTE*, dada a sua compatibilidade com diferentes *hardwares* (PC, IPC e PLC), a independência de sistema operativo no caso do *Codesys Control Win*, assim como a sua compatibilidade com a norma IEC 61131-3 [40]. Esta norma veio normalizar o modo de programar PLCs, estabelecendo linguagens de programação *standard* como ST (*Structured Text*), LD (*Ladder Diagram*), FBD (*Function Block Diagram*) ou SFC (*Sequential Function Chart*) [41].

Os *SoftPLCs* referidos são programáveis através do *software* de desenvolvimento *Codesys V3.5*, que por se tratar de um ambiente de desenvolvimento integrado, permite criar numa mesma aplicação a lógica do controlador e um conjunto de visualizações para interface gráfico. Estas visualizações podem posteriormente ser disponibilizadas numa extensão do *SoftPLC* denominada *TargetVisu*, que pretende emular um painel integrado no próprio controlador [42], resultando numa solução global semelhante às HMIs com PLC integrado referidas no ponto 3.1.1. Para além da *TargetVisu*, é disponibilizada uma outra extensão intitulada *WebVisu*, que, quando incorporada num projeto, ativa o servidor *Web* da *Codesys*, permitindo assim que as visualizações desenvolvidas possam ser acedidas por qualquer *browser* compatível com HTML 5 [42]. A figura 3.5 representa um exemplo de uma visualização criada com o *Codesys V3.5* para uma aplicação no *SoftPLC CODESYS Control Win V3* e disponibilizadas com as referidas extensões.

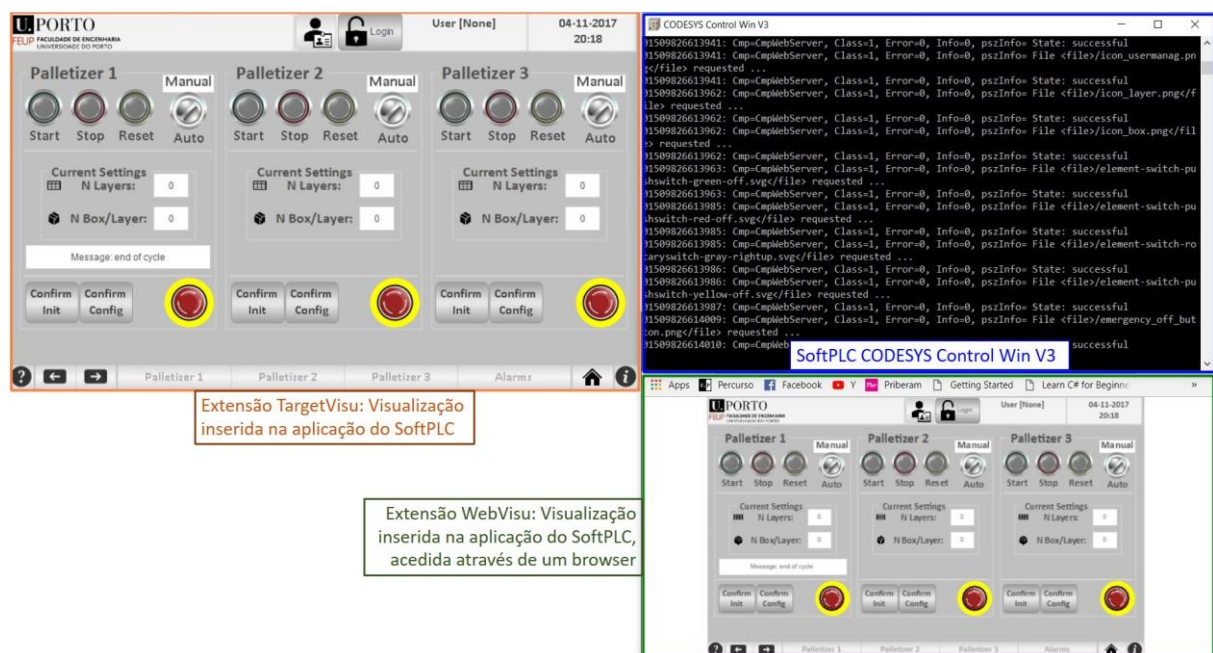


Figura 3.5 – Aplicação de controlo num *SoftPLC CODESYS Control Win V3* com as extensões *TargetVisu* e *WebVisu* para visualização de dados em tempo-real.

Em termos de comunicação, salienta-se a capacidade do *Control Win V3* suportar protocolos como o *Modbus TCP/IP*, *UDP/IP* (variáveis de rede), assim como utilizar um objeto denominado *Data Source Manager* para comunicar ou partilhar variáveis com dispositivos externos, nomeadamente outros controladores ou HMIs, utilizando para isso a porta *Ethernet* do equipamento onde está instalado [43]. Comunicação através do protocolo OPC DA é também possível, uma vez que a *Codesys* possui um servidor OPC (*Codesys OPC Server V3*) cuja interface com o controlador é executada através da aplicação *Gateway V3* [44]. Tanto o servidor OPC, como o *Data Source Manager* (quando o Tipo de *Data Source* é simbólico) requerem a existência de configuração simbólica das variáveis a partilhar, onde são definidos direitos de acesso (leitura, escrita ou ambos) [43].

3.2.2 SoftHMI

Complementando os seus *SoftPLC*, a *3S-Smart Software Solutions GmbH* apresenta ainda a *Codesys HMI* [45], aqui designada como uma *SoftHMI*, dado que pode ser instalada em qualquer PC ou mesmo num painel dedicado, funcionando como uma interface gráfica autónoma sem acesso direto a I/Os [42]. Tal como para o ponto anterior, esta HMI é configurada através do *software* de desenvolvimento *Codesys V3.5* e disponibiliza também a extensão *WebVisu*. A nível de comunicação, aplicam-se todas as vertentes de comunicação acima referidas. A figura 3.6 enquadra a *Codesys HMI* face ao *Control Win V3* com as extensões *TargetVisu* e *WebVisu*.

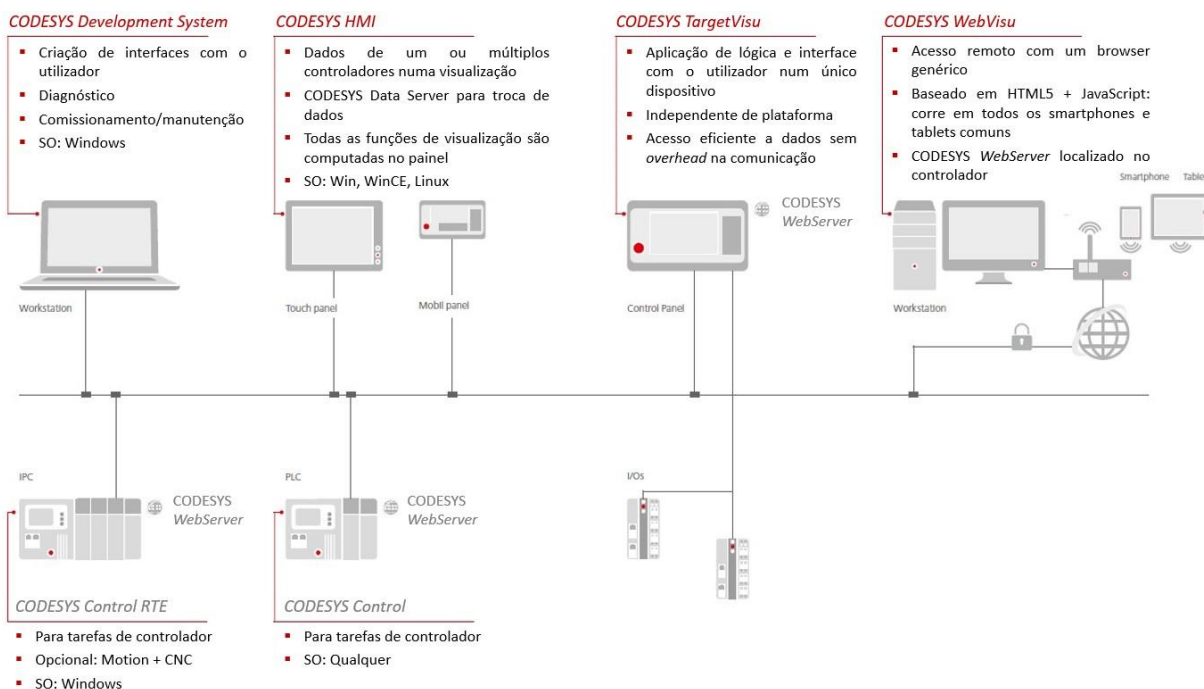


Figura 3.6 – Diferentes formas de visualização de informação com a tecnologia da 3-S Smart Software Solutions GmbH, nomeadamente *CODESYS TargetVisu*, *CODESYS WebVisu* e *CODESYS HMI* [42].

Em termos de funcionalidades disponíveis para as visualizações de interface gráfica, os objetos incluídos no *Codesys* não diferem dos disponibilizados pela generalidade dos fabricantes. As visualizações *Codesys* beneficiam de uma *toolbox* bastante completa: com objetos simples, como formas geométricas e imagens, botões e interruptores, barras, campos de texto, tabelas; assim como com objetos mais avançados, como tabelas de alarmes, gráficos de tendências, *Web browser*, entre outros. Para além destes, o gestor de visualizações permite ainda a inclusão de gestão de utilizadores para uma interface gráfica mais segura [42, 43].

3.3 Mais-valias das Tecnologias Web

De forma a complementar as soluções tecnológicas referidas e proporcionar a mobilidade de informação associada a painéis móveis wireless, contornando o custo elevado associado aos mesmos, surgem novas abordagens que prometem revolucionar o universo da tecnologia de visualização industrial, abrindo a porta para que dispositivos como *tablets* e *smartphones* possam desempenhar um novo papel na mão dos seus utilizadores.

Entre as tecnologias *Web* mais relevantes para a indústria encontram-se as seguintes normas e protocolos da W3C (*World Wide Web Consortium*) [46]:

- **HTML 5 (*Hyper Text Markup Language 5*)**: responsável por estruturar e definir o conteúdo disponível em páginas Web;
- **CSS (*Cascading Style Sheets*)**: responsável pela formatação de páginas Web;
- **JavaScript**: linguagem de programação em HTML, de modo a programar o seu comportamento, criando assim conteúdo HTML dinâmico;
- **XML (*eXtensible Markup Language*)**: semelhante a HTML, no entanto, é uma ferramenta, independente de *software* e *hardware*, desenhada para armazenamento e transporte de dados;
- **HTTP (*HyperText Transfer Protocol*)**: protocolo de comunicação na camada de aplicação que permite a comunicação entre cliente e servidor através de pares de mensagens de pedido/resposta [47].

Existem várias vertentes de implementação, desde soluções nativas na *Web* a soluções nativas ao dispositivo a ser utilizado, no entanto, a maioria destas vertentes têm em comum a arquitetura de implementação cliente-servidor. Entre as modalidades de acesso remoto a PLCs, HMIs e SCADAs, salientam-se as seguintes de maior relevância [48]:

- Servidor: HMI/PLC/SCADA – Cliente: *browser*;
- Servidor: HMI/PLC/SCADA – Cliente: *App* para dispositivo móvel;

- Servidor: HMI/PLC/SCADA – Cliente: *thin client*;
- Servidor: *Cloud based* – Cliente: múltiplos meios de acesso.

As últimas duas modalidades referidas não serão alvo de experimentação prática neste trabalho, no entanto não devem ser negligenciadas, dado o elevado potencial que trazem para “cima da mesa”. Soluções como as oferecidas por [49] disponibilizam uma maior flexibilidade na implementação de tecnologias *Web*, mas requerem um maior conhecimento das mesmas, pelo que é natural que um engenheiro de automação demonstre preferência pelas ferramentas de implementação rápida que serão utilizadas neste trabalho.

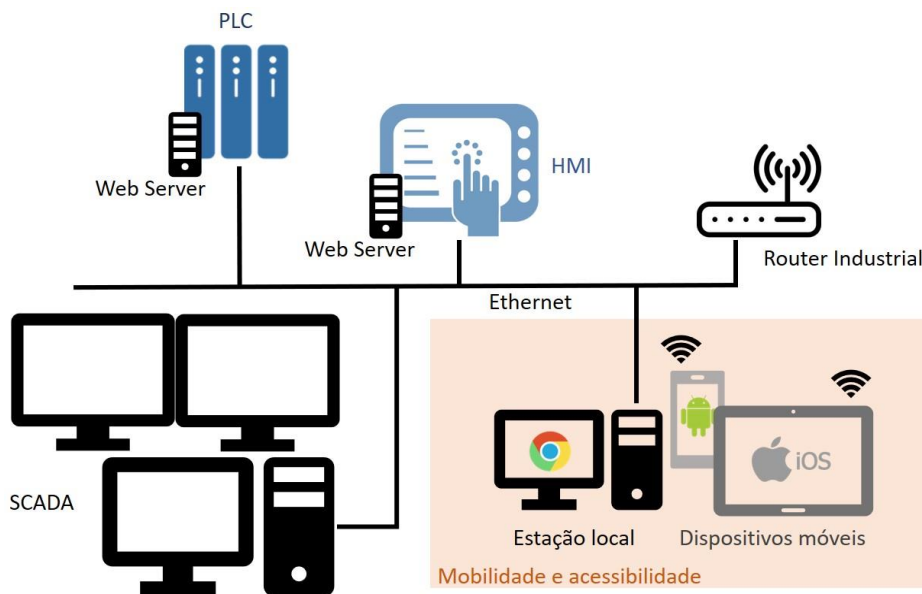


Figura 3.7 – Representação esquemática simples de uma rede industrial.

A figura 3.7, representa, esquematicamente e de um modo simplista, uma rede industrial na qual são implementadas diferentes modalidades da arquitetura cliente-servidor de modo a promover a mobilidade e acessibilidade à informação no chão da fábrica.

Para o caso em estudo, a arquitetura cliente-servidor requer dois intervenientes:

- **o servidor *Web*** – integrado via *software*, quer em PLCs, quer em HMIs, e que tem como função disponibilizar páginas *Web* aos seus clientes, sendo principalmente responsável por processar pedidos recebidos pelo protocolo HTTP, e formular a respetiva resposta. Embora esta seja a definição mais básica de um servidor *Web*, a evolução de páginas que anteriormente eram estáticas para as páginas atualmente dinâmicas, trouxe consigo novas necessidades de resposta para os servidores *Web*. A principal, para o contexto atual é a capacidade de gerar páginas *Web* dinâmicas, incluindo nelas dados em tempo-real.

- **um ou mais clientes** – que assumem diferentes formas, como por exemplo um *browser* ou uma *App* para dispositivo móvel.

Dos tipos de clientes *Web* referidos, o *browser* é o mais transversal a qualquer dispositivo móvel, de qualquer sistema operativo, sendo uma mais valia a sua presença em todos os *smartphones*, *tablets* e PCs, não necessitando a instalação de nenhuma aplicação adicional. Entre os mais conhecidos estão o *Google Chrome*, o *Mozilla Firefox* e o *Safari*. Ainda assim, a forma como a comunicação é estabelecida entre este cliente e o servidor *Web*, terá impacto no desempenho em “tempo-real” desta solução, uma vez que se procuram interfaces gráficos altamente dinâmicos, pelo que deve ser favorecida a troca de dados dinâmicos em detrimento de dados estáticos.

Por sua vez, a utilização de uma *App* como cliente não deixa de ter o seu apelo, como se pode comprovar com uma pesquisa genérica sobre *mobile Apps* para aplicação industrial, tanto na *App Store*, como na *Google Play*. A forte presença de aplicações associadas a produtos de fabricantes como a *Rockwell Automation*, a *Omron*, a *Schneider Electric* e a *Siemens* revela o quão vantajosa esta solução pode vir a ser para a mobilidade e acessibilidade da informação fabril.

Ambas as opções para cliente *Web* apresentam vantagens e desvantagens, pelo que a escolha entre elas terá que ter por base as necessidades específicas de cada caso.

Um aspeto a considerar é a otimização do acesso tendo em conta a dimensão reduzida dos ecrãs de dispositivos móveis face aos ecrãs de PCs. Uma grande quantidade de informação gráfica irá afetar negativamente a velocidade do *download*, e consequentemente o desempenho desta solução como ferramenta de tempo-real, assim como o mau dimensionamento do ecrã tornará o acesso à informação necessária moroso, uma vez que será necessária a utilização de funções como *scroll* [50].

Outro aspeto a considerar será as duas grandes plataformas para dispositivos móveis: *iOS* e *Android*. A linguagem de programação de *Apps* para cada uma destas famílias de dispositivos diferem entre si, pelo que cada fornecedor que queira disponibilizar uma *App* associada ao seu produto terá que escolher uma das plataformas na altura de desenvolvimento, ou então incorrer num maior custo, tanto monetário como temporal, de modo a colocar a sua solução no mercado [50].

No primeiro aspeto a *App* é a clara vencedora, no entanto, o facto de o *browser* não ser condicionado pelo sistema operativo, nem pelo *browser* utilizado graças à especificação atualmente normalizada HTML 5, faz dele mais vantajoso no segundo ponto.

Uma solução que conjuga o melhor dos dois mundos será uma aplicação baseada na *Web*, desenvolvida recorrendo à referida especificação HTML 5, contornando desta forma as limitações impostas pelas linguagens de programação nativas [48].

3.4 Soluções tecnológicas a explorar na dissertação

Uma vez analisadas as soluções tecnológicas para visualização industrial descritas neste estado da arte, descrevem-se e justificam-se em seguida as opções tomadas a nível tecnológico para a implementação do trabalho proposto nesta dissertação.

Dado que se pretende a implementação simultânea de aplicações de controlo e de visualização de cenários industriais, optou-se por uma solução que permitisse a integração simultânea de ambas, reduzindo o esforço de implementação global da solução. Neste contexto, escolheu-se o ambiente de desenvolvimento integrado *Codesys V3.5* da alemã *3S-Smart Software Solutions GmbH*, e os respetivos *SoftPLC* e *SoftHMI*, *Codesys Win V3* e *Codesys HMI*. Este *software* não só está em conformidade com a IEC 61131-3, norma de maior relevância no que diz respeito à programação de PLCs, como é um exemplo de *software* independente cuja tecnologia é usada como suporte aos próprios ambientes de desenvolvimento de vários OEMs de automação industrial, como por exemplo *ABB*, *Bosch Rexroth*, *Eaton*, *Schneider Electric*, *WAGO*, *Festo*, entre muitos outros. Trata-se, portanto, de uma tecnologia representativa de um vasto leque de soluções para visualização de dados em tempo-real, justificando a sua exploração mais aprofundada no capítulo seguinte. Mais informação relativa a este *software* pode ser encontrada em www.codesys.com.

No entanto, e de modo a obter resultados qualitativos do trabalho desenvolvido é vantajoso e enriquecedor explorar um outro ambiente de desenvolvimento integrado. Para tal optou-se pelo *SIMATIC STEP 7* e pelo *SIMATIC WinCC*, integrantes do *TIA Portal* da também alemã *Siemens*, e os respetivos PLC e painel embutido, *S7-1200* e *SIMATIC KTP600*. Aqui a escolha apoia-se não só na conformidade com a IEC 61131-3, mas também na cota de mercado detida pela *Siemens*, que possui uma das maiores bases de clientes na automação industrial e, para além disto, como se verificou no ponto 2.1.2, encontra-se ativamente investida em ultrapassar os desafios futuros para visualização de dados em tempo-real. Por essa razão, o capítulo 5 dedica-se a explorar de forma mais profunda as funcionalidades proporcionadas pela tecnologia da *Siemens*.

3.4.1 Solução tecnológica da 3S-Smart Software Solutions GmbH

A primeira solução consistirá na utilização do *software* de desenvolvimento *Codesys V3.5*, na sua versão *SP10 Patch 1*, a mais recente até à data de início deste trabalho, e com algumas funcionalidades de grande interesse para o âmbito desta dissertação.

O *SoftPLC Codesys Control Win V3* será utilizado para controlar um cenário industrial virtual desenvolvido no *software* de emulação *Factory I/O*. Por sua vez, serão utilizados os objetos *TargetVisu* e *WebVisu* para respetivamente integrar no controlador uma interface gráfica para o operador e ativar o servidor *Web*, permitindo assim o acesso à interface criada através de um *browser*. Tratar-se-á então de um acesso remoto do tipo: servidor *Web* PLC e cliente *browser*.

Já a *SoftHMI Codesys HMI* tomará o papel de uma HMI autónoma especificamente desenhada com as necessidades de supervisão e manutenção em mente. Neste dispositivo o servidor *Web* também estará ativo de modo a permitir que o engenheiro de manutenção ou supervisão tenha acesso a esta interface em qualquer sítio da fábrica.

Por fim utilizar-se-ão dispositivos móveis, como um *tablet* ou um *smartphone*, para que, através de um *browser*, seja possível aceder remotamente às interfaces de manutenção e supervisão.

A figura 3.8 exemplifica a arquitetura e comunicações estabelecidas para a implementação de visualizações industriais com a tecnologia *Codesys*. Tanto o controlador como o cenário virtual encontram-se no mesmo *hardware*, sendo a comunicação entre os dois realizada através do servidor OPC do *Codesys*. Já a comunicação entre o controlador e a HMI de supervisão/manutenção será realizada através de uma conexão ponto-a-ponto sobre *Ethernet*, facilitada pelo objeto *Data Source Manager* do *Codesys*.

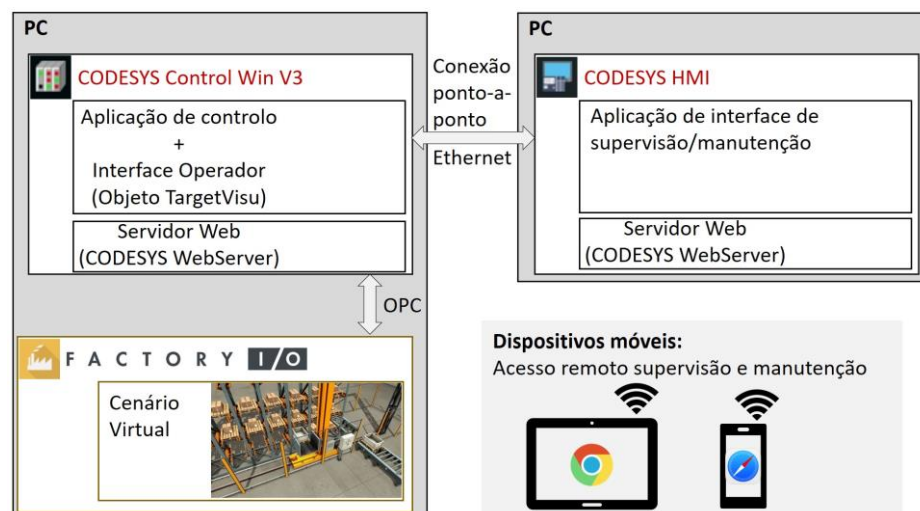


Figura 3.8 – Arquitetura da solução a implementar com a tecnologia Codesys V3.5.

3.4.2 Solução tecnológica da Siemens

A segunda solução apoiar-se-á na tecnologia da *Siemens*, nomeadamente, o PLC S7-1200 CPU 1214C e o painel *SIMATIC KTP600*, assim como os seus *softwares* de edição *STEP 7* e *WinCC (TIA Portal)*.

A escolha do painel *SIMATIC KTP 600*, pertencente à gama básica de painéis da *Siemens*, pretende demonstrar a versatilidade de funções que este disponibiliza, ainda que seja considerado para aplicações simples. A escolha de um PLC S7-1200 assenta no facto de este pertencer à gama básica de PLCs da *Siemens*, o que se adequa para a aplicação em causa, assim como no facto de possuir todas as funcionalidades desejadas, incluindo um servidor *Web* integrado, o que complementa de forma ideal o painel escolhido.

O PLC S7-1200 controlará o cenário industrial do *Factory I/O*, comunicando com ele através de um protocolo proprietário da *Siemens* sobre *Ethernet*. Isto é possível uma vez que o *Factory I/O* possui um driver para comunicação com o S7-1200. Para além disso, o controlador terá o seu servidor *Web* ativo, no qual se encontrarão páginas *Web* a serem utilizadas por engenheiros de manutenção ou de supervisão.

O painel *SIMATIC KTP600* será o painel do operador, comunicando com o controlador através do mesmo protocolo utilizado para o ambiente virtual. Por fim, os dispositivos móveis serão responsáveis pelo acesso remoto às páginas no servidor *Web*, quer seja via *browser* ou via *Apps* desenvolvidas pela *Siemens*. Tratar-se-á então de um acesso remoto do tipo: servidor *Web* PLC e cliente *App* ou *browser*.

Na figura 3.9 observa-se a arquitetura e comunicações estabelecidas para a implementação de visualizações industriais com a tecnologia *Siemens*.



Figura 3.9 – Arquitetura da solução a implementar com a tecnologia da Siemens.

3.5 Síntese

Este capítulo descreveu o estado da arte das tecnologias mais relevantes para visualização industrial, tanto em termos de *hardware* como de *software*. Foram apreciadas as características e funcionalidades de equipamento e ferramentas disponibilizadas por diferentes fabricantes, salientando a forma como cada um consegue ou não proporcionar uma comunicação fluída de informação a nível fabril.

Constatou-se a pertinência da aplicação das tecnologias *Web* no meio industrial, utilizando arquiteturas cliente-servidor, como forma de potenciar o fluxo de informação na fábrica para todos os intervenientes: operador, engenheiro de manutenção e supervisor.

Por fim, e com base na análise realizada, enunciaram-se de forma justificada as soluções tecnológicas a implementar nos próximos capítulos. Uma primeira, com base na tecnologia da *3S-Smart Software Solutions GmbH* e a segunda alicerçada na gigante de automação industrial, *Siemens*.

4 DESENVOLVIMENTO DE SOLUÇÕES DE VISUALIZAÇÃO EM CODESYS

O relato do trabalho experimental desenvolvido nesta tese encontra-se dividido em duas partes distintas, dissociando deste modo as duas soluções tecnológicas escolhidas para o desenvolvimento de interfaces de grande versatilidade, acessibilidade e mobilidade, com recurso a visualizações *Web*. A primeira parte, trabalhada e exposta no presente capítulo, centra-se na utilização de recursos de programação *Codesys V3.5*. A segunda, apoiada na tecnologia *Siemens* é desenvolvida no próximo capítulo.

Inicialmente, é definido o cenário industrial a recriar no *software* de simulação *Factory I/O* – que passa por uma célula de paletização – sendo feita uma breve contextualização do problema e uma descrição sumária das máquinas envolvidas. De seguida, são especificadas as interações e interfaces a desenvolver para os três níveis de utilizadores considerados. Posteriormente, é introduzido o ambiente de programação *Codesys*, sendo exposta a modelação do controlo dos paletizadores e exploradas mais a fundo as suas funcionalidades para a visualização do cenário industrial implementado. Na etapa seguinte são desenvolvidas e testadas as interfaces para o cenário escolhido.

4.1 Aplicação de suporte e teste

O ponto de partida desta experiência foi definir e criar uma aplicação industrial com relevância prática que fosse alvo da interação com três tipos de utilizadores: operador local, técnicos de manutenção e encarregados de supervisão. Para tal utilizou-se o *Factory I/O*, um *software* criado pela empresa portuguesa *Real Games* que permite a construção e simulação dos mais variados ambientes industriais, facilitando a sua utilização como instrumento para a aprendizagem de tecnologias de automação industrial. O facto de este utilizar um motor físico de tempo real resulta em cenários extremamente realistas, de fácil interação entre o utilizador e a simulação. Por esse motivo, e aliando a este a disponibilização gratuita do *software* no laboratório de Sistemas Lógicos da FEUP, o *Factory I/O* foi a ferramenta utilizada para a criação e emulação dos cenários industriais utilizados ao longo deste trabalho. Toda a documentação desta aplicação pode ser encontrada em www.factoryio.com.

Tendo como base as aplicações industriais relevantes enunciadas e descritas no capítulo 2, optou-se pela utilização de uma aplicação de paletização, que proporciona a complexidade necessária em termos de comando e de monitorização, sendo um bom caso de estudo para a demonstração das funcionalidades da tecnologia *Codesys*.

4.1.1 Aplicação de paletização

A aplicação escolhida consiste numa célula com três paletizadores, observável na figura 4.1, que conseguem individualmente produzir paletes com diferentes configurações, isto é, diferente número de camadas e número de caixas por camada, conforme os parâmetros definidos. Uma vez que se trata de três equipamentos com idênticas funcionalidades, basta descrever o respetivo comportamento para um deles.



Figura 4.1 – Célula de paletização desenvolvida no Factory I/O.

O paletizador possui duas entradas de materiais, uma para caixas, no 1º piso e uma para paletes vazias, no piso 0. No 1º piso é sobre uma placa móvel que ocorre a conformação das camadas de caixas a colocar na paleta, uma camada de cada vez. Simultaneamente, no piso 0 são alimentadas paletes no elevador do paletizador, também uma de cada vez. Estando a plataforma elevatória ocupada por uma paleta esta é elevada até ao 1º piso onde uma camada de caixas é depositada através da abertura da placa móvel. Para a deposição de mais camadas a plataforma elevatória executa um movimento descendente “passo-a-passo”, permitindo que a placa móvel feche e uma nova camada de caixas seja formada e posteriormente depositada. Este comportamento ocorre ciclicamente até o número de camadas desejado seja atingido. Por fim, uma vez concluída a tarefa, a plataforma elevatória regressa ao piso 0 onde o produto final é descarregado para um tapete de saída, de modo a ser eliminado do ambiente de simulação; e onde uma nova paleta vazia é carregada, repetindo o comportamento acima descrito.

Na figura 4.2, subdivide-se o equipamento de paletização em quatro subsistemas, sendo desta forma possível compartimentar as funções do equipamento. Distinguem-se então os seguintes subsistemas:

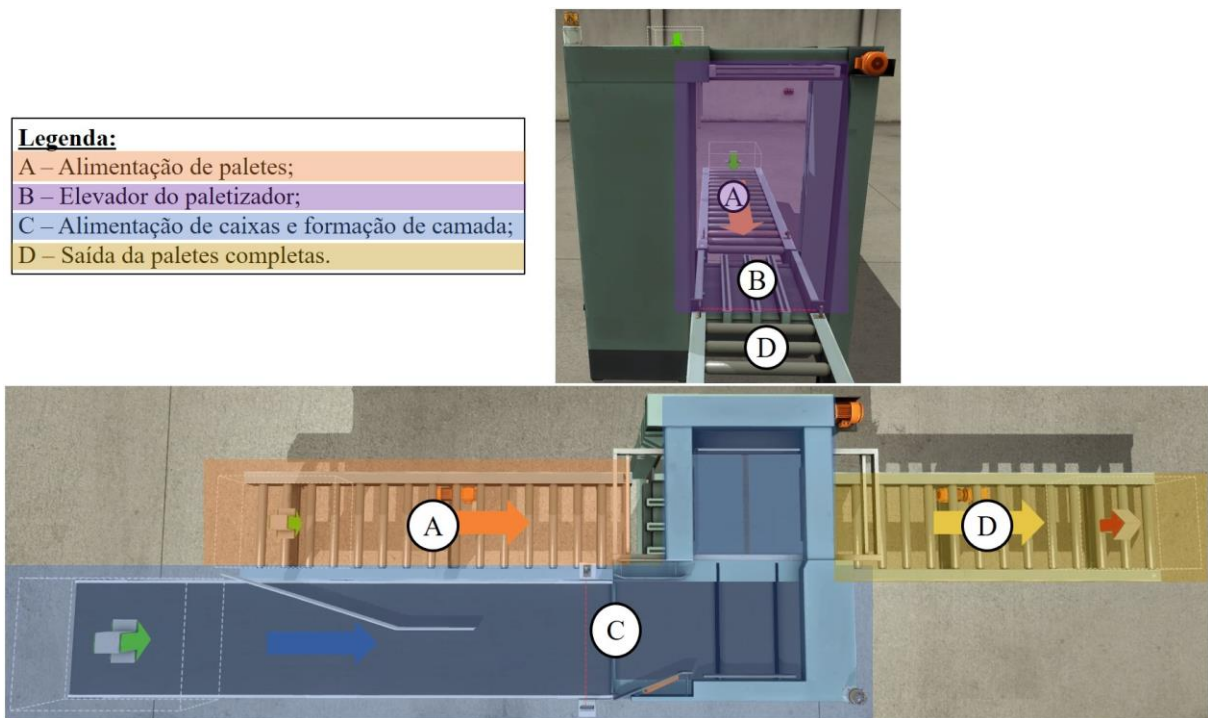


Figura 4.2 – Subsistemas de um equipamento de paletização.

- **A – Alimentação de paletes** – constituído por um elemento emissor de paletes vazias, um tapete transportador de rolos e dois sensores retrorrefletivos à entrada e saída do tapete. Este subsistema tem como principal função manter uma alimentação contínua de paletes vazias até ao subsistema B.
- **B – Elevador de paletes** – constituído por uma plataforma elevatória, dotada de movimento ascendente e descendente, podendo esse movimento ser executado de uma forma contínua ou então “passo-a-passo”. Esta plataforma elevatória possui por sua vez um elemento de transferência por correntes, o que permite a carga e descarga de paletes do subsistema A para o subsistema B e do subsistema B para o subsistema D, respetivamente. A nível sensorial, possui dois sensores capacitivos de curto alcance, o que permite aferir a presença ou não de uma paleta na plataforma, e possui ainda um sensor de movimento para identificar se a plataforma se encontra parada ou em movimento ascendente/descendente.
- **C – Alimentação de caixas e formação de camada** – constituído por um elemento emissor de caixas; um tapete transportador de tela com dois sensores retrorrefletivos à entrada e saída deste; um atuador que altera a orientação das caixas em 90°; um tapete de tela onde são agrupadas as caixas a ser empurradas pelo elemento *pusher* para uma

placa móvel com movimento de translação (abertura/fecho). Para além destes, tem ainda elementos fixadores que são atuados quando uma camada é completada. Alguns dos elementos aqui referidos podem ser observados na figura 4.3.

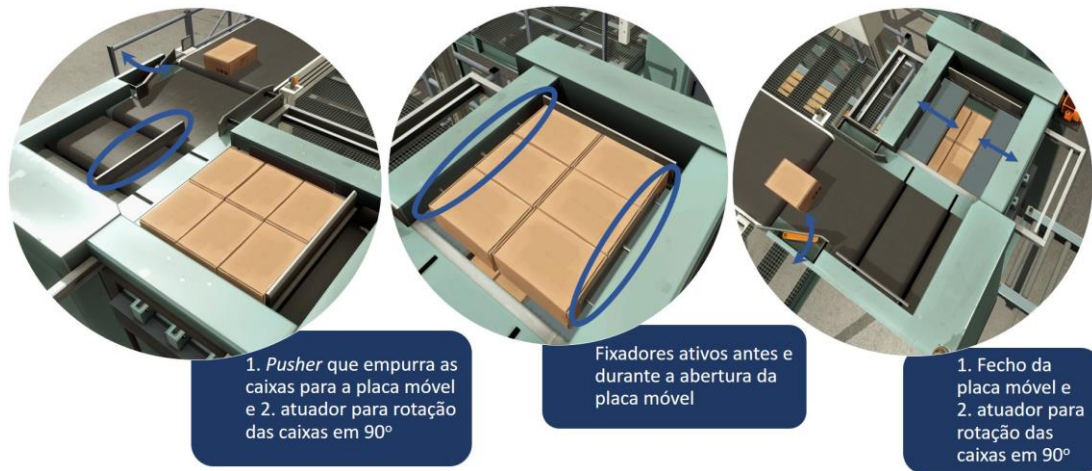


Figura 4.3 – Detalhes do subsistema C – Alimentação de caixas e formação de camada.

- **D – Saída de paletes completas** – constituído por um tapete transportador de rolos, um sensor retrorrefletivo à entrada deste tapete e um elemento eliminador do produto final. Tem como principal função retirar o produto final, isto é, uma paleta completa, do subsistema B.

Para além dos subsistemas referidos, foi ainda criado um quadro elétrico de controlo no ambiente virtual para cada um dos paletizadores da figura 4.1. Este permite lançar comandos básicos de operação do equipamento, facilidade importante na ausência das interfaces gráficas ainda a ser desenvolvidas. As tabelas 4.1 e 4.2 representam as variáveis de interface do controlador com o cenário virtual, sendo que a primeira descreve as variáveis de controlo do paletizador e a segunda as variáveis de operação. As variáveis de entrada e saída no controlador consistem, respetivamente, em sinais vindos de sensores e para atuadores na tabela 4.1; e sinais vindos de botões e para sinalizadores na tabela 4.2.

Tabela 4.1 – Variáveis de interface do controlador com o cenário virtual: controlo do paletizador.

Subsistema A				Subsistema C			
Variáveis de entrada (sensores)		Variáveis de saída (atuadores)		Variáveis de entrada (sensores)		Variáveis de saída (atuadores)	
Sensor palletAtEntry	Bool (N)	Tapete palletFeeder	Bool	Sensor boxAtEntry	Bool (N)	Tapete boxFeeder	Bool
Sensor palletEmitted	Bool (N)	palletEmitter	Bool	Sensor boxEmitted	Bool (N)	Tapete loadBelt	Bool
Subsistema B				pusherLimit	Bool	push	Bool
Variáveis de entrada (sensores)		Variáveis de saída (atuadores)		clamped	Bool	turn	Bool
Sensor elevatorMoving	Bool	elevatorUp	Bool	plateLimit	Bool	clamp	Bool
Sensor palletLoaded	Bool	elevatorDown	Bool			openPlate	Bool
		moveToLimit	Bool			partEmitter	Bool
		loadPallet	Bool				
				Subsistema D			
Variáveis de entrada (sensores)		Variáveis de saída (atuadores)		Variáveis de entrada (sensores)		Variáveis de saída (atuadores)	
				Sensor palletAtExit	Bool (N)	Tapete exitConveyor	Bool
						remover	Bool

(N) Lógica invertida

Tabela 4.2 – Variáveis de interface do controlador com o cenário virtual: operação do paletizador.

Painel de Controlo Factory I/O			
Variáveis de entrada (botões)		Variáveis de saída (sinalizadores)	
bStart	Bool	lightStart	Bool
bStop	Bool (N)	lightStop	Bool
bReset	Bool	lightReset	Bool
bPause	Bool	lightPause	Bool
bEmerg	Bool (N)	warningLight	Bool
Auto	Bool	countPalletsIn	Int
Manual	Bool	countPalletsOut	Int
		countBoxesIn	Int
		countBoxBuffer	Int
		countPalletBuffer	Int

(N) Lógica invertida

4.1.2 Cenários de falha a considerar

Para além das variáveis de interface entre controlador e máquina, são necessárias variáveis que permitam a interface entre as HMIs e o controlador. Estas terão como principal função flexibilizar o modo de funcionamento do equipamento, sem que seja necessário intervir diretamente no código da aplicação de controlo. Para que seja possível uma produção flexível de paletes é necessário que o operador possa introduzir parâmetros de configuração das paletes a construir. Mas, para além do modo de funcionamento automático, será conveniente um modo de operação manual, onde o operador tem acesso direto ao comando das variáveis de saída (atuadores), não ficando o funcionamento do equipamento limitado às rotinas de controlo automático implementadas.

Embora o objetivo de um equipamento automático seja a sua operação continua sem intervenção do operador, é inevitável a ocorrência de situações anómalas durante o seu funcionamento. Dada essa inevitabilidade, é importante que essas situações sejam devidamente sinalizadas, de modo a que o operador possa ter uma intervenção manual rápida sobre as mesmas, retornando o equipamento ao seu funcionamento normal.

Uma implementação inicial da aplicação do paletizador permitiu identificar as zonas de maior fragilidade funcional do equipamento, isto é, as zonas onde podem ocorrer mais falhas de equipamento, sendo então necessária uma sinalização adequada das mesmas na forma de alarmes. Entre as situações encontradas salientam-se as seguintes:

- Transição da paleta do subsistema A (alimentação de paletes) para o subsistema B (elevador de paletes): paleta fica presa entre os dois sistemas, resultando num encravamento do funcionamento do sistema.

- Atraso na elevação da paleta pelo subsistema B (elevador de paletes): resulta numa falha de coordenação do subsistema B com o subsistema C (alimentação de caixas e formação de camada).
- Número de caixas no segundo tapete de tela do subsistema C ultrapassa o valor desejado;
- *Pusher* do subsistema C não consegue cumprir a totalidade do curso devido a encravamento de uma caixa.

Contudo, as situações acima referidas não são as únicas merecedoras de alarme, uma vez que, numa situação real, o desgaste de atuadores e sensores pode levar à avaria extemporânea destes, o que constitui de igual forma uma situação anómala. De modo a incluir esta possibilidade no caso de estudo, serão introduzidas na simulação do *Factory I/O* “avarias”, tais como perda de sinal de um sensor ou falha de energização de um atuador.

4.2 Especificação de interfaces

Uma vez contextualizada a aplicação do paletizador, torna-se necessário especificar as funções que as HMIs, isto é, as interfaces com o operador, engenheiro de manutenção e engenheiro de supervisão, devem ser capazes de cumprir. Com base nos problemas descritos no ponto anterior, enunciam-se em seguida as especificações das interfaces a desenvolver, de modo a solucionar esses problemas.

Como especificação base, todas as interfaces devem exigir autenticação com utilizador e *password* para acesso às mesmas. Deve ser utilizado o mesmo *template*, isto é, mesma paleta de cores base e mesma disposição de elementos padrão, como dados do utilizador atual, data e hora, logotipo da empresa e menu de navegação. Desta forma garante-se uma navegação consistente nas interfaces a desenvolver.

4.2.1 Especificação da interface de operação

No que diz respeito à especificação de *hardware*, a interface do operador tem de ser local à célula de paletização, podendo-se optar por um PC industrial com painel embutido, onde o *SoftPLC CODESYS Control Win V3* seria instalado. A nível experimental esse papel foi assumido por um PC do laboratório de Sistemas Lógicos da FEUP.

Quanto às restantes especificações, a interface do operador deve cumprir as seguintes funções:

- ✓ Permitir acesso local e remoto;
- ✓ Em caso de acesso local e remoto simultâneo, apenas um utilizador deve ter privilégio de comando sobre a estação, sendo este à partida do utilizador local. No entanto, deverá ser possível que este privilégio de comando passe para o utilizador remoto, mediante o conhecimento e aceitação do utilizador local (por exemplo, a definição de um modo de manutenção);
- ✓ Permitir que o operador selecione o modo de funcionamento do equipamento: manual ou automático;
- ✓ No caso de operação manual, permitir:
 - Acionar individualmente os diferentes atuadores do sistema;
 - Acionar separadamente os subsistemas A, B, C ou D;
- ✓ No caso de operação automática, permitir:
 - Definir os parâmetros de funcionamento da máquina (nº de camadas, nº de caixas por camada, nº de paletes a construir);
 - Confirmar ou corrigir os parâmetros inseridos;
 - Dar ordens de iniciar (*Start*), de paragem (*Stop*) em qualquer momento da produção, de purga do equipamento, isto é, esvaziamento da máquina (*Reset*), de paragem de emergência (*Emerg*), e ainda um comando de pausa (*Pause*);
 - Alterar o número de paletes no lote durante a produção;
 - Descartar caixas/paletes e compensar os parâmetros de produção em conformidade;
- ✓ Disponibilizar *feedback* constante quanto ao estado de funcionamento do equipamento, recorrendo também à geração de eventos;
- ✓ Permitir a monitorização da produção: dados atuais sobre o número de caixas, camadas e paletes já executadas, previsão temporal para conclusão da ordem de produção, etc;
- ✓ Utilização de gráficos para visualização da evolução da produção;
- ✓ Informar o utilizador quanto à existência de situações anómalas de funcionamento de uma forma clara e hierarquizada (definição de prioridade de alarmes) e permitir a aceitação de alarmes;

- ✓ Permitir filtrar os alarmes consoante classe ou grupo;
- ✓ Permitir acesso a histórico de situações anómalas ocorridas.

Para cumprir as especificações apresentadas desenvolveram-se vários ecrãs, que se relacionam entre si através de um menu de navegação. Isto permite um acesso hierarquizado à informação e funcionalidades de comando, tal como a árvore de ecrãs da figura 4.4 pretende demonstrar. Sempre que possível recorreu-se à utilização de elementos simbólicos, normalizados segundo as normas “ISO 7000 Graphical symbols for use on equipment” e “IEC 60417 Graphical symbols for use on equipment”. A utilização destes símbolos permite ao utilizador uma rápida perceção quanto à informação que cada ecrã disponibilizará, dado o reconhecimento universal destes símbolos gráficos.

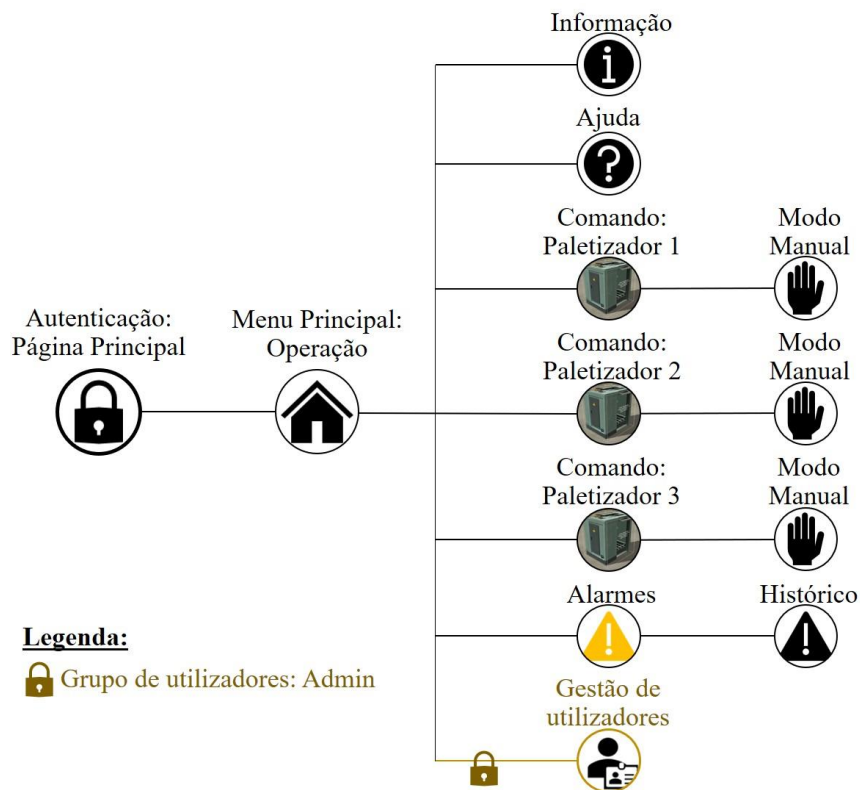



Figura 4.4 – Árvore de ecrãs da interface do operador.

No entanto, existem ecrãs que não são acessíveis através do menu de navegação, uma vez que podem aparecer sobre qualquer ecrã como resposta a certos eventos e alarmes ocorridos no sistema. Estes podem ser meramente informativos (eventos), chamando a atenção do utilizador para uma dada situação e pedindo confirmação por parte deste, ou então, podem conter informação específica à resolução de uma situação anómala (alarmes), podendo até reencaminhar o utilizador para um ecrã específico. Na tabela 4.3, descrevem-se alguns eventos que levam ao aparecimento de janelas *popup*, assim como o tipo de janelas (informativa ou alerta, pedido de confirmação ou de dados) e as ações que podem ser tomadas a partir destas.

Tabela 4.3 – Eventos e alarmes que ativam janelas popup e respetiva possibilidade de ações, interface do operador.

	Eventos	Tipo de Popup	Ações
	Start	Pedido de confirmação	Aceitar ou rejeitar os parâmetros atuais do paletizador.
	Pause	Informativa	Fechar ecrã ou retomar a produção.
	Stop fim de ciclo	Pedido de confirmação	Esvaziar equipamento ou iniciar nova ordem de produção.
	Emergência	Informativa	Aceitar e/ou fechar a janela.
	Alarmes	Alerta	Confirmar alarmes e/ou fechar janela
	Login	Pedido de dados	Inserir dados de utilizador e password, confirmar ou cancelar.

4.2.2 Especificação da interface de manutenção

A interface do engenheiro de manutenção é constituída em termos de *hardware* por qualquer dispositivo móvel (*tablet* ou *smartphone*) ou PC com um *browser* instalado. Estes equipamentos são capazes de aceder a um PC industrial não local à estação de paletização, onde seria instalada a *SoftHMI Codesys* HMI. No caso de estudo, esse papel foi assumido por um PC do laboratório de Sistemas Lógicos da FEUP.

A nível de um engenheiro de manutenção, a interface deve desempenhar as seguintes funções:

- ✓ Permitir acesso local e remoto;
- ✓ Em caso de acesso local e remoto simultâneo, apenas um utilizador deve ter privilégio de comando sobre a estação, que por *default* deve ser o utilizador que se encontra autenticado em primeiro lugar;
- ✓ Acesso a modelos comportamentais do equipamento, por exemplo Grafsets, representados por elementos gráficos que acompanhem a sua evolução de uma forma dinâmica;
- ✓ Sinalização clara de anomalias do equipamento no modelo comportamental, para uma rápida intervenção do utilizador;
- ✓ Acesso alarmes de sinalização de anomalias de funcionamento do equipamento e respetivo histórico;
- ✓ Permitir registar evolução de variáveis em gráficos de tendências e aceder o histórico dessa evolução para um espaço de tempo definido;
- ✓ Permitir comando da máquina aquando a instalação da mesma e aquando serviços de manutenção (modo de comando manual);

- ✓ Acesso ao tempo de serviço dos diversos elementos do equipamento (atuadores /sensores) e aferição da necessidade de intervenções de manutenção preventiva;
- ✓ Acesso a manuais do equipamento ou a conteúdos audiovisuais que auxiliem no procedimento de manutenção.

4.2.3 Especificação da interface de supervisão


A interface do engenheiro de supervisão é materializada da mesma forma que a interface do engenheiro de manutenção, devendo desempenhar as seguintes funções:

- ✓ Permitir acesso local e remoto;
- ✓ Em caso de acesso local e remoto simultâneo, apenas um utilizador deve ter privilégio de comando sobre a estação, que por *default* deve ser o utilizador que se encontra autenticado em primeiro lugar;
- ✓ Acesso alarmes de sinalização de anomalias de funcionamento do equipamento e respetivo histórico;
- ✓ Permitir registar evolução de variáveis em gráficos de tendências e aceder o histórico dessa evolução para um espaço de tempo definido;
- ✓ Permitir o comando da máquina quando o operador não tem capacidade de resolver anomalias de funcionamento de maior gravidade;
- ✓ Acesso a registos de eventos do funcionamento do equipamento (atuais e históricos), assim como da atividade de autenticação dos utilizadores das diferentes interfaces;
- ✓ Acesso a dados estatísticos como indicadores de desempenho do equipamento (KPIs): tempo de produção, tempo de manutenção e tempo de paragem.

A figura 4.5 representa a árvore de ecrãs desenvolvidos para a interface de manutenção e supervisão, sendo as funcionalidades disponibilizadas restringidas consoante o nível de autenticação utilizado.

Tal como para a interface do operador, existem janelas *popup* para a interface de manutenção e de supervisão. Na tabela 4.4, descrevem-se alguns eventos que levam ao aparecimento dessas janelas, assim como o tipo de janelas (informativa, pedido de confirmação, alerta) e as ações que podem ser tomadas a partir destas.

Tabela 4.4 – Eventos e alarmes que ativam janelas popup e respetiva possibilidade de ações, interface de manutenção e de supervisão.

	Eventos	Tipo de Popup	Ações
	Emergência	Informativa	Aceitar e/ou fechar a janela.
	Alarmes	Alerta	Confirmar alarmes e/ou fechar janela
	Login	Pedido de dados	Inserir dados de utilizador e password, confirmar ou cancelar.

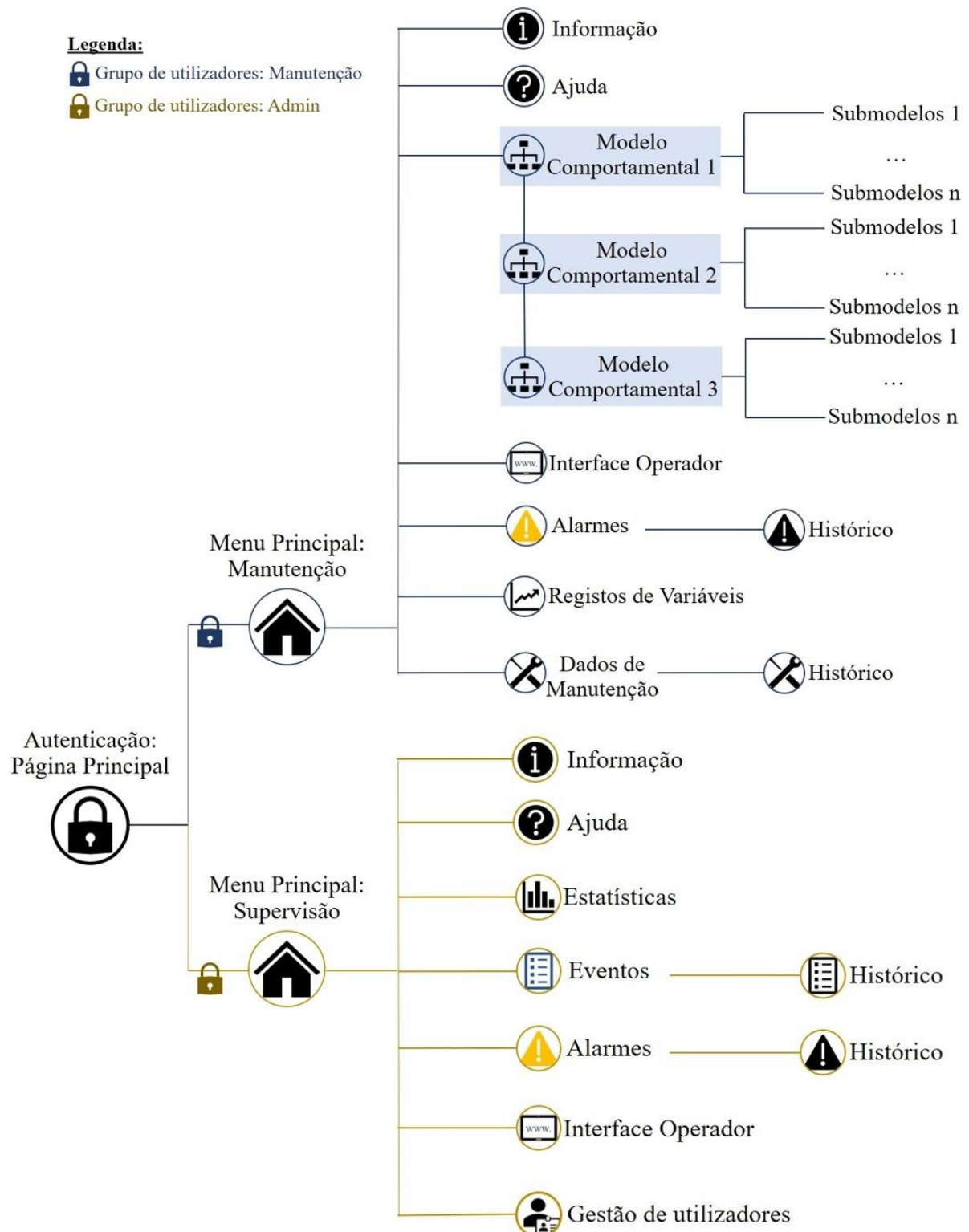


Figura 4.5 – Árvore de ecrãs da interface de manutenção e de supervisão.

4.3 Codesys

O *Codesys* é uma ferramenta muito moderna e extremamente poderosa no que à automação industrial diz respeito. Com particular interesse para o trabalho a desenvolver, salientam-se as linguagens de programação *Structured Text* (ST) e *Continuous Function Chart* (CFC), ambas utilizadas na programação do controlador, e a primeira na programação de algumas funcionalidades das visualizações. Também a facilidade em criar bibliotecas que podem ser reutilizadas em vários projetos foi uma vantagem para criar elementos gráficos que permitissem cumprir algumas das especificações referidas. Para além disto, as extensas bibliotecas disponibilizadas à partida no *Codesys* e o ambiente de desenvolvimento *user friendly* permitem a qualquer engenheiro de automação uma rápida implementação de visualizações, reduzindo a necessidade de conhecimentos de linguagens de programação de alto nível como C/C++, C#, entre outras.

Os modelos de controlo são uma etapa importante deste trabalho, dado que sem eles não existiria um modo de funcionamento automático do cenário virtual. No entanto, a sua descrição detalhada não constitui uma parte fundamental da temática deste projeto, pelo que os modelos de controlo desenvolvidos recorrendo à linguagem GRAFCET constam no Anexo A.1 devidamente explicados.

4.3.1 Deteção de situações anómalas

Com os cenários de falha acima referidos em mente, foram criados blocos capazes de detetar as situações de falha supramencionadas. Estes consistem na sua maioria em blocos que monitorizam os sinais de atuadores ou sensores com base nas etapas atuais dos modelos comportamentais que lhes estão associadas.

Por exemplo, se o alimentador de paletes estiver numa etapa onde é acionado o tapete de entrada e este não estiver ativo, daqui resulta um alerta quanto à falta de energização do atuador.

Outros blocos, do tipo *watchdog*, isto é, temporizações, permitem aferir se o paletizador permanece no mesmo conjunto de etapas demasiado tempo, sendo possível desta maneira alertar o operador para situações como: uma caixa mal posicionada que impede o avanço completo do *pusher* aquando a execução de uma camada; o facto de tanto uma camada estar pronta, como a paleta estar pronta a receber e nada acontecer; entre outras.

4.3.2 Visualizações: objetos e funcionalidades

O ambiente de desenvolvimento do *Codesys* permite a utilização de vários objetos e bibliotecas integradas de modo a implementar as especificações típicas de uma interface gráfica. Nesta secção sumarizam-se os objetos e funcionalidades de maior relevância e que suscitaram maior interesse:

- **Gestão de utilizadores** (figura 4.6): implementada através da biblioteca “*VisuUserManagement*” 3.5.10.0, permite uma integração rápida no projeto da especificação que exige autenticação por utilizador e password, permitindo a definição de grupos de utilizadores e consequentemente uma hierarquia de acesso à interface;

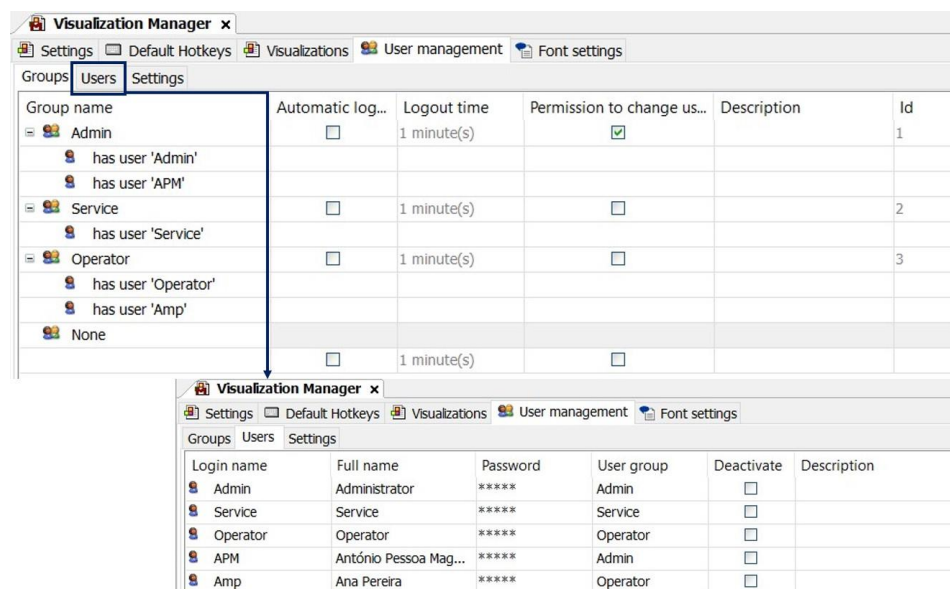


Figura 4.6 – Configuração de gestão de utilizadores em Codesys.

- **Objetos básicos** (figura 4.7): estes elementos distinguem-se dos restantes pela sua versatilidade de aplicação, podendo ser utilizados para obter gráficos mais avançados. Apesar de serem intitulados objeto básicos permitem todo o tipo de configurações desde aparência e visibilidade, até resposta a *inputs* do utilizador. Salienta-se a importância do objeto “*frame*”, que permite a inclusão de visualizações já criadas numa nova;

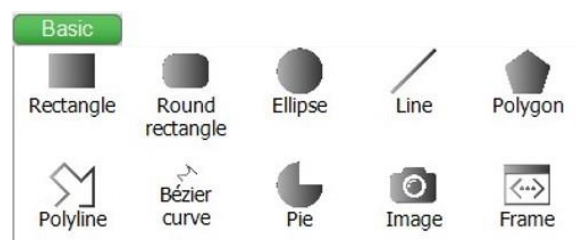


Figura 4.7 – Objetos básicos da toolbox do Codesys, de salientar o retângulo, a imagem e a frame.

- **Objetos de controlo** (figuras 4.8): estes objetos têm propósitos concretos já definidos, sendo a configuração de cada um focada na funcionalidade que vai desempenhar. Estes elementos servem como elementos de *input* e/ou *output* de informação para o utilizador;

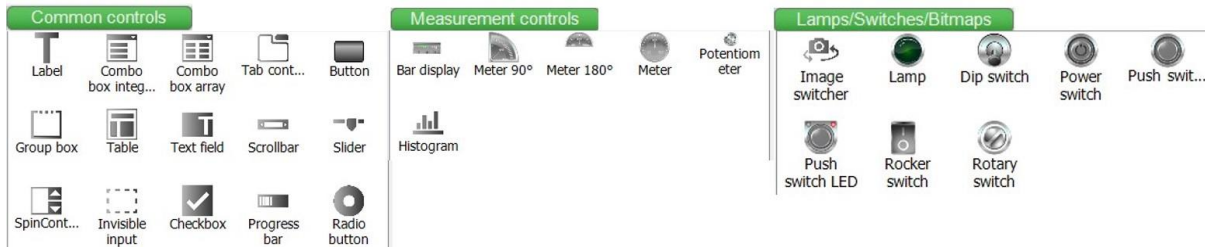


Figura 4.8 – Objetos de controlo da toolbox do Codesys.

- **Objetos complexos** (figura 4.9): estes permitem a visualização de dados complexos, executando funções pré-configuradas em bibliotecas no Codesys. Entre eles destacam-se a tabela de alarmes, o registo de tendências e o *Webbrowser*.



Figura 4.9 – Objetos complexos da toolbox do Codesys, de salientar a tabela de alarmes, o registo de tendências e o Web browser.

- **Configuração de alarmes** (figura 4.10): permite a configuração de alarmes discretos, analógicos e de resposta a eventos, sendo estes atribuídos a um dado grupo de alarmes, e a uma dada classe (erro, aviso, evento, informação). A classe de alarmes define qual o método de confirmação de alarme exigido, assim como a prioridade dos alarmes pertencentes aquela classe, se devem ou não ser arquivados, quais as ações de notificação a tomar (mudar uma variável, executar dado programa, etc.) e ainda as opções de aparência na tabela de alarmes.

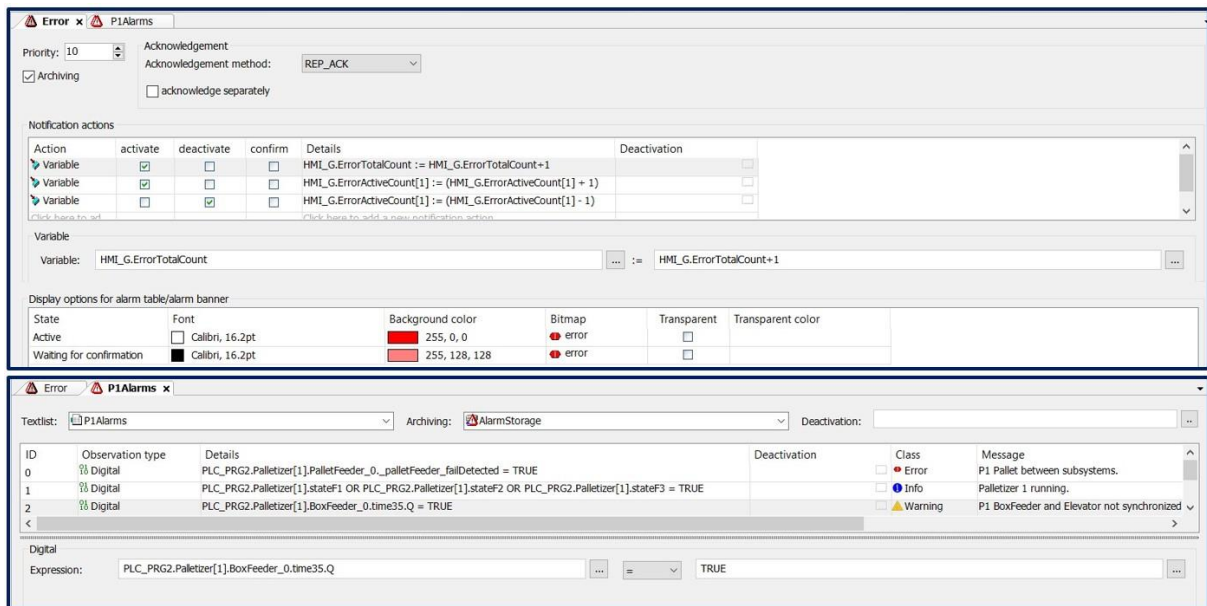


Figura 4.10 – Configuração de alarmes em Codesys.

4.4 Programação de interfaces

Uma vez estabelecida a hierarquia de controlo e analisados os objetos de visualização disponíveis no *Codesys*, iniciou-se a implementação das especificações acima descritas, de modo a obter as interfaces de controlo e monitorização do ambiente virtual.

Na figura 4.11 encontra-se o ecrã de autenticação para as duas interfaces desenvolvidas, onde é descrito de modo sumário como é permitido o acesso ao ecrã principal.

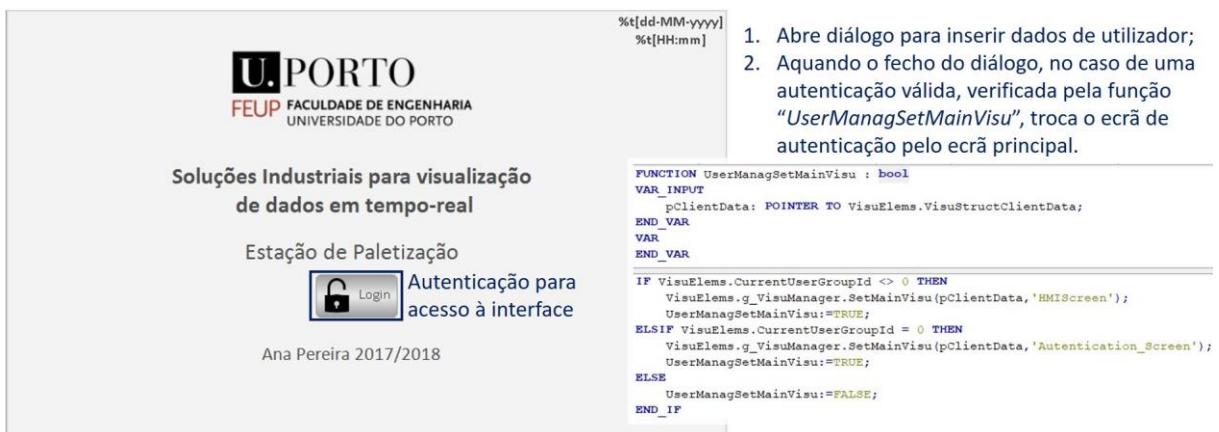


Figura 4.11 – Ecrã de autenticação.

Por sua vez, nas figuras 4.12 e 4.13, encontram-se os *templates* criados para a interface do operador e de manutenção/supervisão, respetivamente. Estes dividem-se em três áreas: o cabeçalho, onde se encontra o logótipo da empresa, data e hora e tudo o está relacionado com a gestão de utilizadores; o painel de navegação, que permite aceder aos diferentes ecrãs das árvores representadas nas figuras 4.4 e 4.5; e a área principal onde se encontra o ecrã selecionado pelo utilizador no painel de navegação.

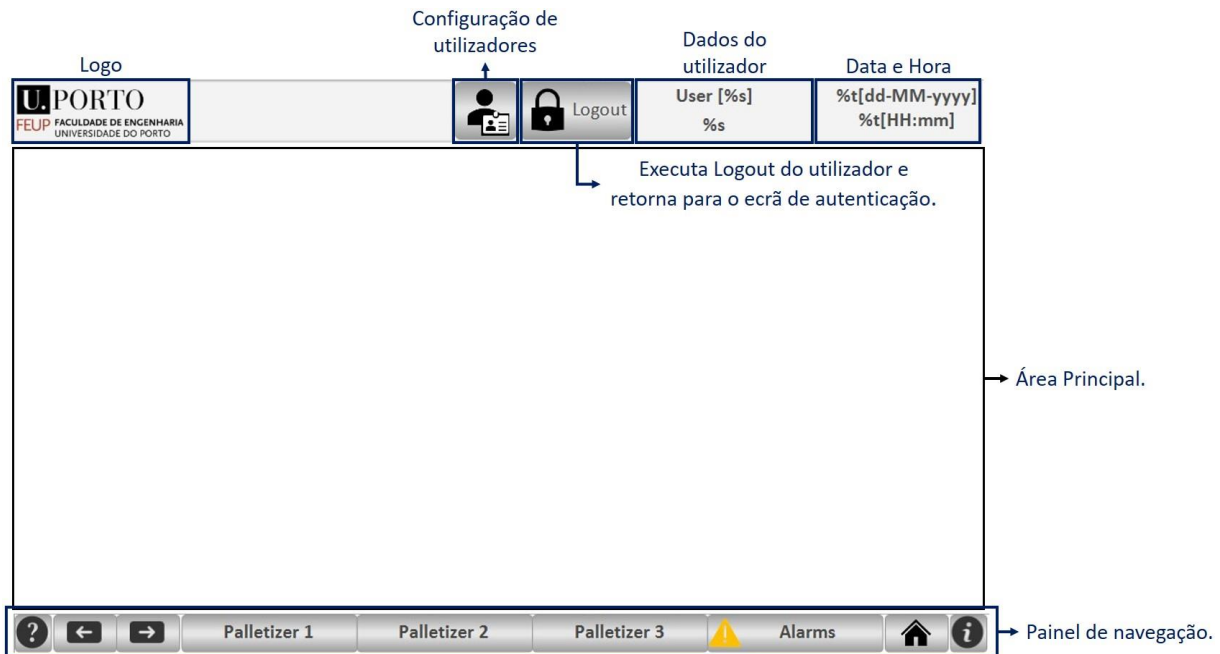


Figura 4.12 – Template para a interface do operador.

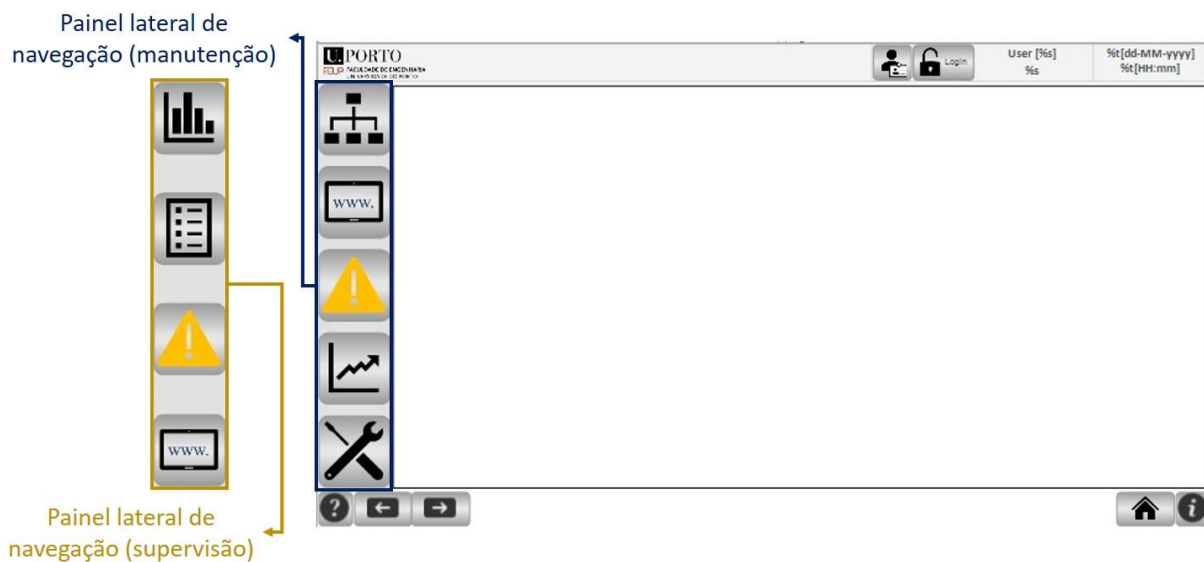


Figura 4.13 – Template para a interface de manutenção e de supervisão.

No caso das interfaces de manutenção e de supervisão, optou-se por adicionar um painel de navegação lateral, mantendo apenas no painel inferior os botões de navegação sequencial, de ajuda, de informação e de menu inicial (*home screen*). O painel de navegação permite que utilizadores diferentes, por exemplo, um com acesso local e outro com acesso remoto, acessem a ecrãs diferentes simultaneamente, facilitando assim monitorização remota, sem interferência com as operações de comando do utilizador local. As setas do painel de navegação permitem uma navegação sequencial da interface, criando um histórico de navegação, que é reiniciado quando é premido o botão “Home”.

4.4.1 Interface do operador

A partir de agora serão apresentados os diferentes ecrãs que constam na árvore de ecrãs da interface do operador.

A figura 4.14 apresenta a página Menu principal: operação, nela encontram-se todos os controlos necessários para colocar qualquer um dos paletizadores em funcionamento. Neste ecrã são também visíveis o número de erros, avisos e informações ativas ou por confirmar.

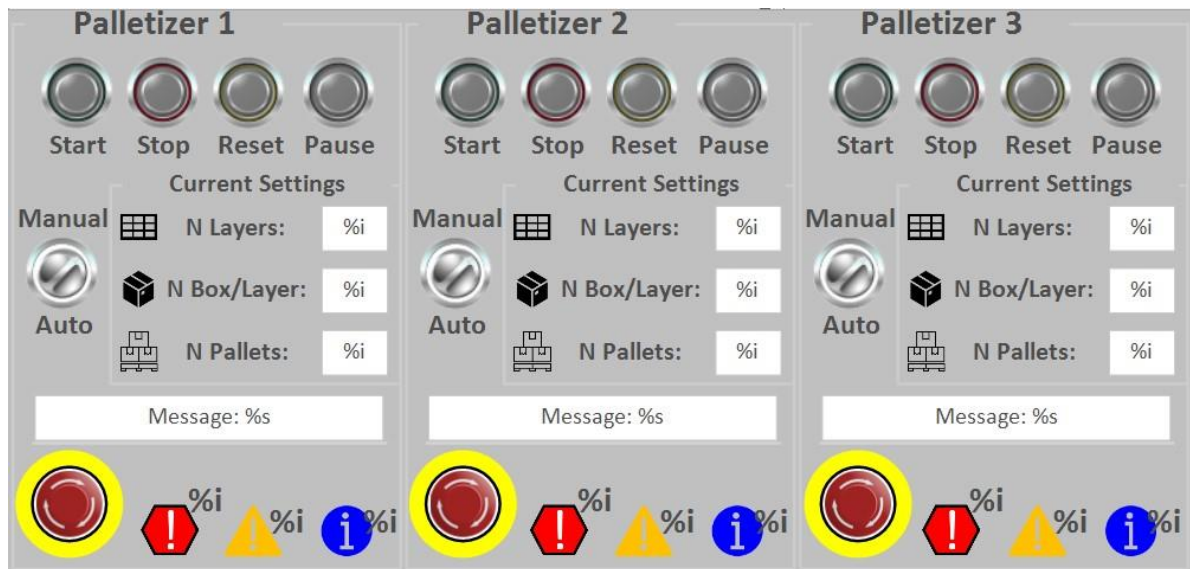


Figura 4.14 – Ecrã Menu Principal (“Home Screen”) da interface de operação.

A figura 4.15 apresenta a página de controlo automático e monitorização dedicada ao paletizador 1, servindo esta de exemplo para as duas restantes. Para além dos controlos necessários, esta página permite ao operador observar a evolução no número de caixas e de paletes a serem executadas, assim como proceder ao ajuste de parâmetros de produção durante a mesma.

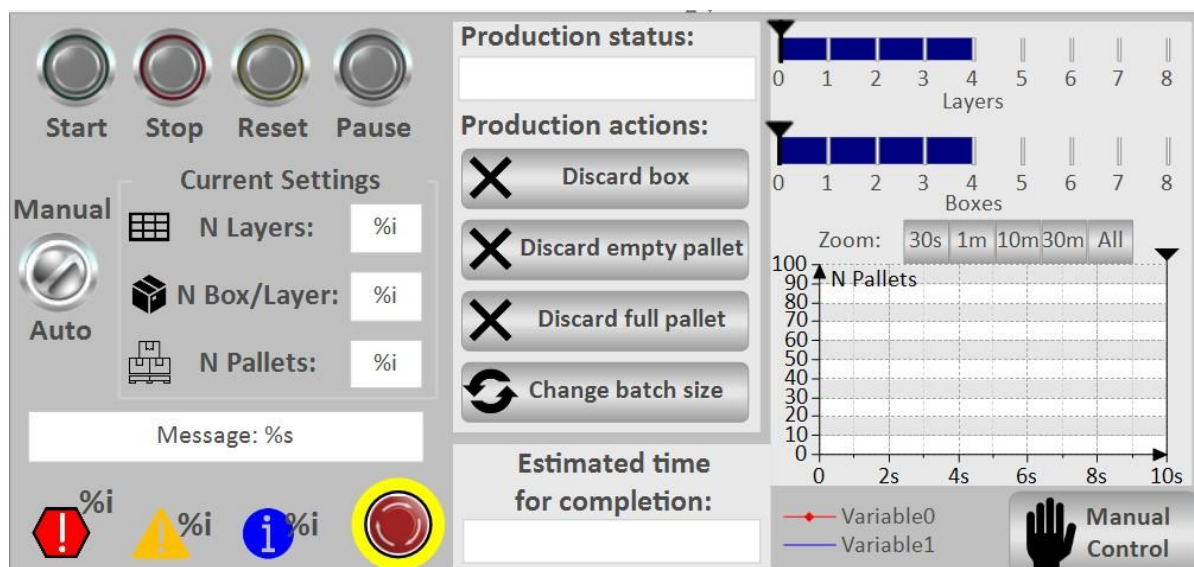


Figura 4.15 – Ecrã Controlo automático do paletizador 1 da interface de operação.

A figura 4.16 representa a página de Controlo manual do paletizador 1, servindo esta novamente de exemplo para as duas restantes.

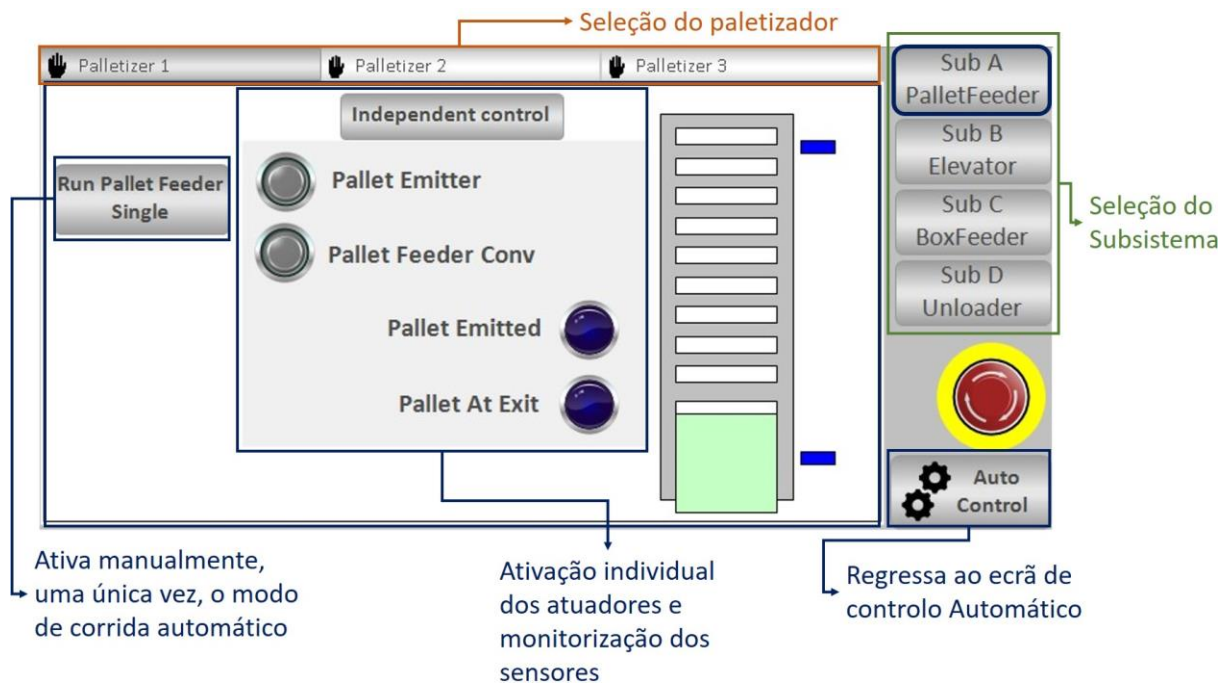


Figura 4.16 – Exemplo de ecrã de Controlo manual do paletizador 1 da interface de operação.

Por sua vez, a figura 4.17 apresenta o ecrã de Alarmes, onde é possível selecionar os alarmes a visualizar consoante o grupo e a classe. A partir deste ecrã é também possível aceder ao histórico dos alarmes.

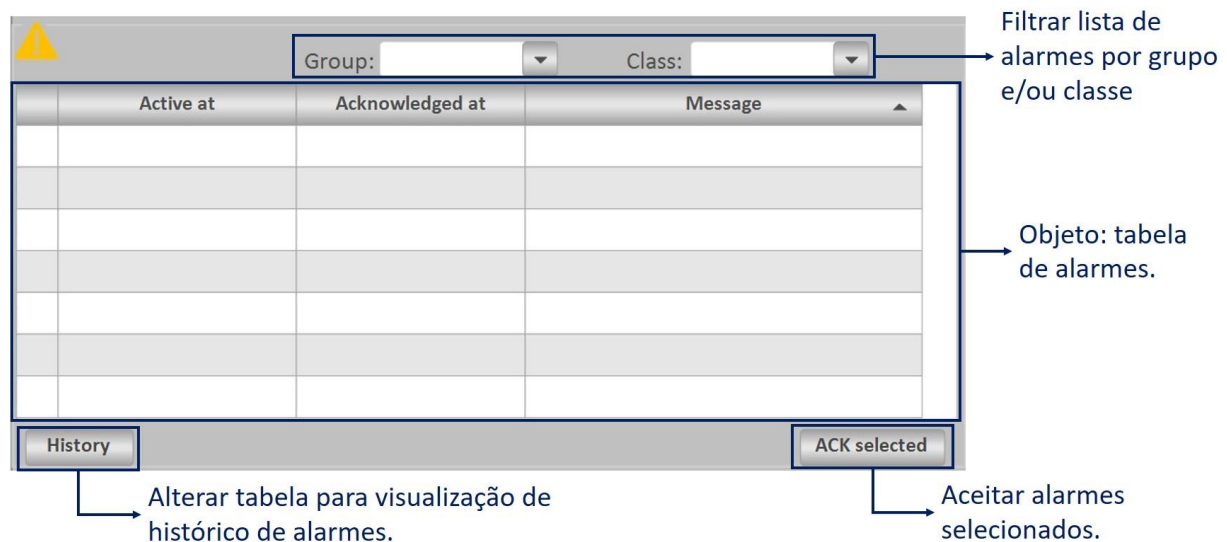


Figura 4.17 – Ecrã de Alarmes da interface de operação.

Por fim, a figura 4.18 apresenta os ecrãs de Informação e de Ajuda. O ecrã de gestão de utilizadores, apenas acessível a níveis de autenticação de Administrador será abordado na próxima secção.



Figura 4.18 – Ecrã de Informação (esquerda) e ecrã de Ajuda (direita) da interface de operação.

4.4.2 Interface de manutenção e supervisão

Aqui serão apresentados primeiro os ecrãs desenvolvidos para a interface de manutenção, seguindo-se os ecrãs desenvolvidos para a interface de supervisão.

No caso da interface de manutenção, o Menu Principal coincide com o ecrã do Modelo Comportamental do paletizador 1. Já no caso da interface de supervisão, o Menu Principal coincide com o ecrã de estatísticas.

Na figura 4.19 está representado um ecrã do Modelo Comportamental do paletizador 1, servindo de exemplo para os outros dois.

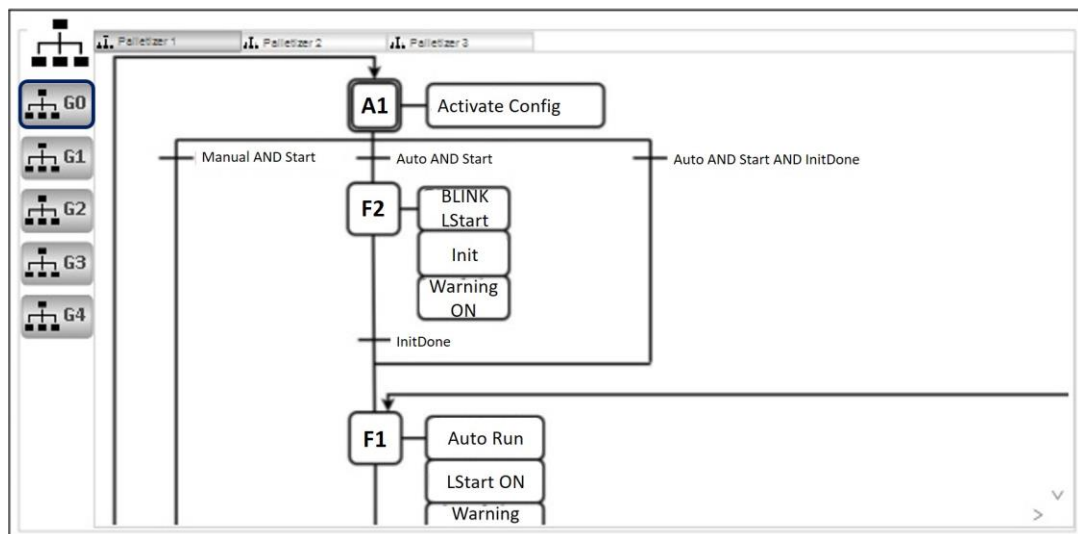


Figura 4.19 – Ecrã do Modelo comportamental do paletizador 1, da interface de manutenção.

Este ecrã foi concebido utilizando uma biblioteca criada no *Codesys*, com elementos que permitissem montar de forma recorrente e simples os modelos de comportamento do equipamento. Este ecrã permite acompanhar em tempo real a evolução dos diferentes Graficets do paletizador 1, pretendendo-se que com isto seja mais fácil aferir as causas de anomalias.

A figura 4.20 apresenta o ecrã de registo de tendências de variáveis. Nele é possível observar a evolução temporal de variáveis em gráficos de tendências, assim como observar em formato de tabela qual o valor atual de dada variável e quando foi a última vez que este sofreu alteração.

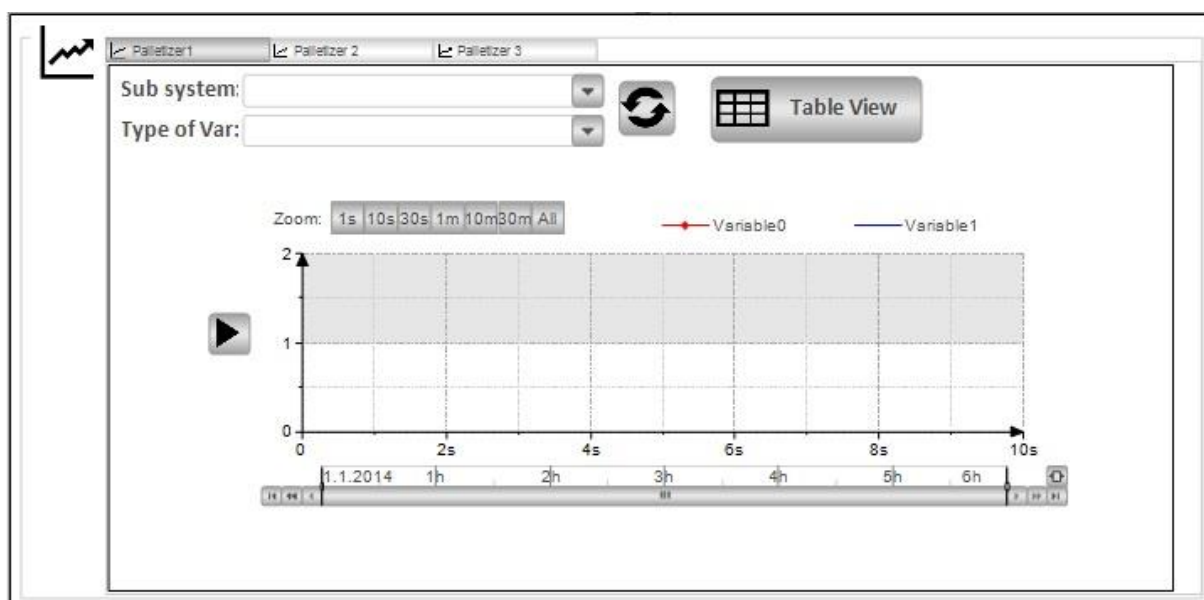


Figura 4.20 – Ecrã de Registo de variáveis da interface de manutenção.

A figura 4.21 apresenta o ecrã de Manutenção, sendo que nele é possível ter acesso a dados como o tempo de serviço ou número de ciclos efetuados de diversos elementos dos equipamentos, assim como número de avarias registadas de cada um.

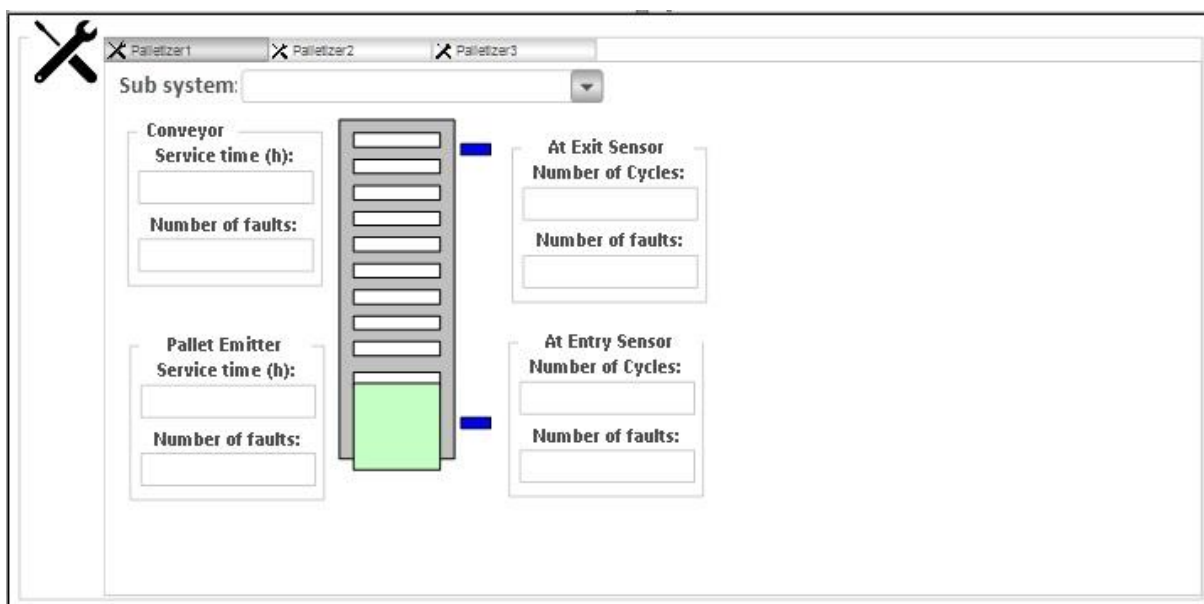


Figura 4.21 – Ecrã de Manutenção da interface de manutenção.

A figura 4.22 demonstra os ecrãs de Informação e de Ajuda da interface de manutenção, sendo que neste último, é possível aceder a meios, como manuais de utilizador ou *data sheets* de elementos do equipamento.



Figura 4.22 – Ecrã de Informação (esquerda) e ecrã de Ajuda (direita) da interface de manutenção.

A figura 4.23 apresenta o ecrã de Estatísticas da interface de supervisão. Nele observam-se dados estatísticos diários como tempo de produção, paragem ou de manutenção, tal como uma análise à quantidade de avarias ocorridas por equipamento.

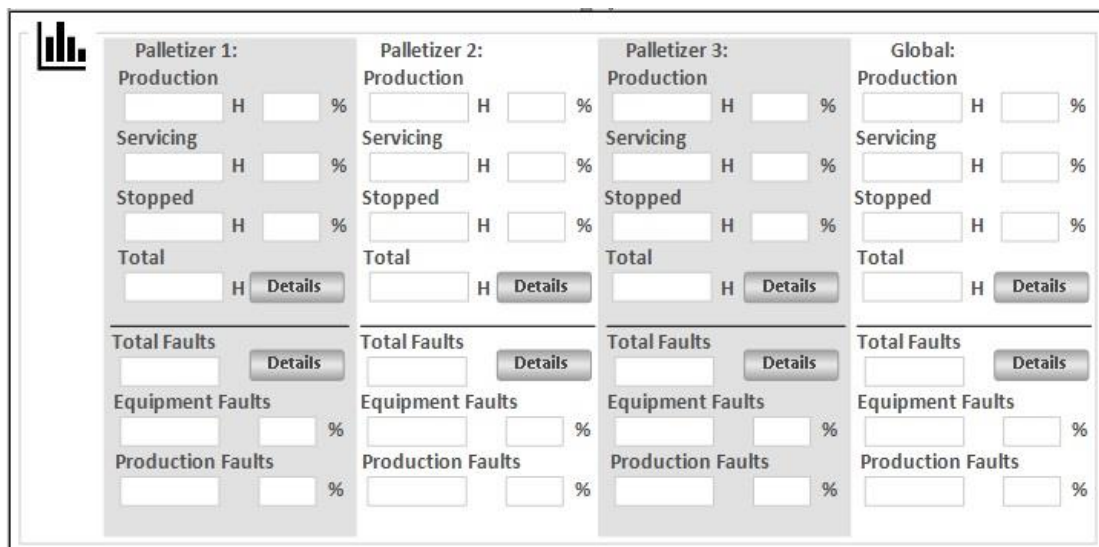


Figura 4.23 – Ecrã de Estatísticas da interface de supervisão.

A figura 4.24 permite o acesso ao registo de Eventos que podem ser filtrados por equipamento.

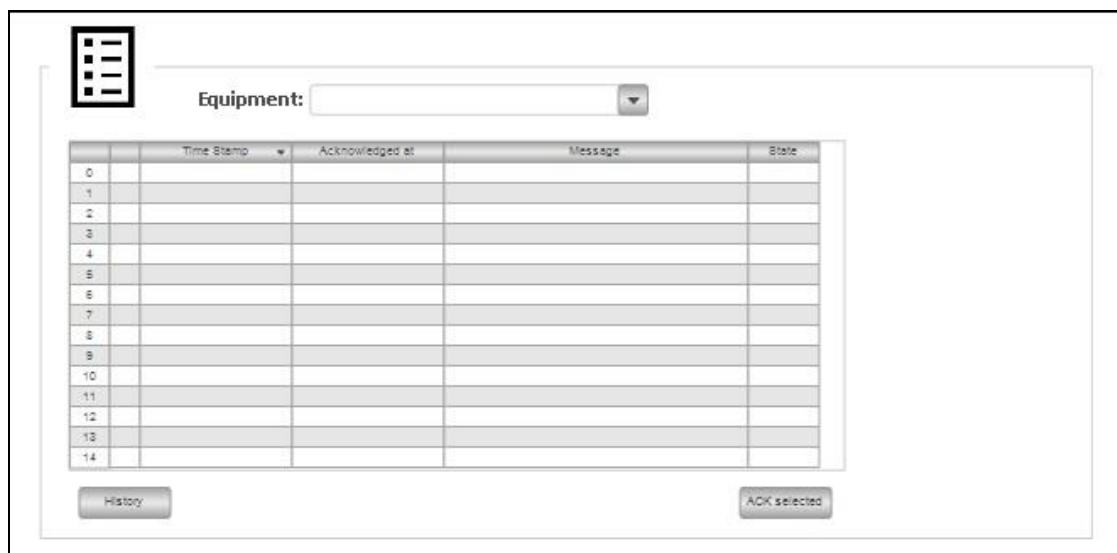


Figura 4.24 – Ecrã de Eventos da interface de supervisão.

A figura 4.25 representa os ecrãs de Informação e de Ajuda da interface de supervisão.



Figura 4.25 – Ecrã de Informação (esquerda) e ecrã de Ajuda (direita) da interface de supervisão.

Por fim, a figura 4.26 apresenta a janela de gestão de utilizadores onde um utilizador com perfil de administrador pode adicionar/remover utilizadores e ainda alterar *passwords*.

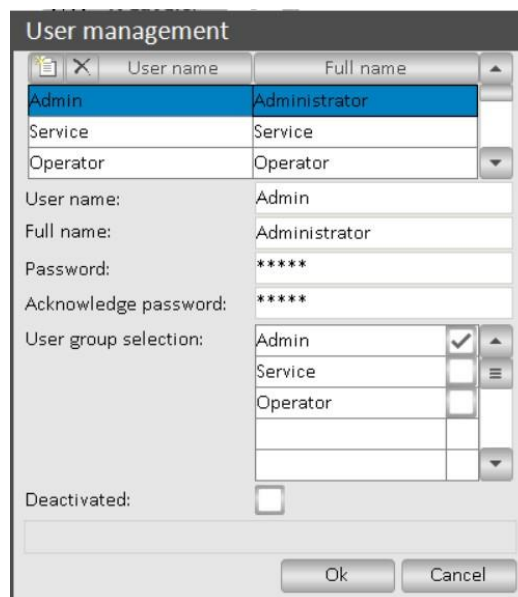


Figura 4.26 – Janela de gestão de utilizadores, da interface de operação e da interface de supervisão.

Os ecrãs de Interface do Operador de ambas as interfaces de manutenção e de supervisão, que permitem aceder remotamente à interface do operador, não estão representados dado que não são nada mais do que a interface do operador supra descrita. O mesmo acontece com os ecrãs de Alarmes, que são semelhantes ao exposto na figura 4.17.

4.5 Discussão dos resultados

O trabalho desenvolvido e exposto neste capítulo permitiu, com alguma dificuldade, e com algumas condicionantes, cumprir as especificações enunciadas no ponto 4.2. Em particular, no caso da interface de manutenção e de supervisão, onde as funcionalidades especificadas foram implementadas de forma segmentada, uma vez que o projeto completo resultava numa sobrecarga de processamento da tarefa responsável pelas visualizações, assim como da tarefa responsável pela comunicação entre *SoftHMI* e *SoftPLC*.

De facto, uma das grandes dificuldades na implementação das visualizações em *Codesys* foi a instabilidade do *software*, uma vez que, para projetos formulados de forma semelhante, o comportamento das funcionalidades inseridas, nomeadamente registo de tendências ou de alarmes, era diferente, dando bons resultados numa ocasião e falhando noutra sem razão aparente. Para além disso foram ainda encontradas algumas incoerências no comportamento de dados objetos, consoante as opções de escala seleccionadas para a visualização.

Foi nestes aspetos que se sentiu uma dificuldade acrescida em resolver os problemas encontrados, dada a falta de documentação de suporte, em concreto no que diz respeito aos códigos de erros reportados no registo de diagnóstico do *SoftPLC*. Ao longo do projeto foram procuradas razões fundamentadas para o comportamento demonstrado junto da *3S-Smart Software Solutions*, no entanto, apesar da simpatia de quem nos atendeu, nem sempre foram obtidas as respostas pretendidas.

Apesar da grande diversidade de funcionalidades que se conseguiu implementar com sucesso, a crescente complexidade de objetos gráficos introduzidos, assim como a crescente necessidade de troca de dados entre *SoftPLC* e *SoftHMI* causou algumas falhas a nível de desempenho, nomeadamente a cumprir os tempos de ciclo impostos. O esforço de processamento de interfaces gráficas desenvolvidas em *Codesys* com o intuito de cumprir todas as especificações referidas, contrasta com interfaces desenvolvidas em linguagens de alto nível, fora do domínio dos engenheiros de automação industrial. Torna-se assim evidente que a abordagem *user friendly* deste tipo de ambientes de desenvolvimento de interfaces gráficas, limita a implementação de funcionalidades mais avançadas, tanto em quantidade como em qualidade, tendo em conta os requisitos de tempo-real exigidos.

De forma a contrariar as falhas de desempenho, aumentaram-se os tempos de ciclo tanto da tarefa responsável pelas visualizações, como da tarefa responsável por estabelecer a comunicação com a interface de supervisão e de manutenção. No entanto, a certo ponto, este aumento deixou de ser uma solução viável, sentindo-se um atraso considerável na resposta da interface face aos *inputs* introduzidos pelo utilizador.

Para além das especificações iniciais, chegaram a ser exploradas funcionalidades *multiTouch* e de *design* responsivo disponibilizadas no *Codesys*. No entanto, não se justificou a implementação das mesmas, dada a sobrecarga de processamento já reportada. Do mesmo modo, existem funcionalidades, disponibilizadas no *Codesys*, que não foram abordadas neste projeto, uma vez que não se coadunavam com a aplicação em causa. Entre elas destacam-se a

gestão de receitas, a utilização de componentes *ActiveX*, assim como objetos ligados a aplicações CNC, como desenho 3D de trajetórias e painel de controlo apropriado.

Por fim, relativamente ao acesso remoto via *browser*, pode-se afirmar que este foi implementado com sucesso, salientando-se a facilidade de obter páginas *Web* dinâmicas a partir *Codesys*, sem qualquer conhecimento base de HTML ou *JavaScript*. Isto constitui uma solução satisfatória para aceder a dados complexos do processo sem esforço adicional por parte do programador, no entanto, é de referir que esta solução poderá não estar otimizada em termos de recursos exigidos ao servidor *Web*.

Em suma, o *Codesys* apresentou-se como uma ferramenta de grande flexibilidade, que permitia uma conceção de visualizações “à medida”, tendo para isso um custo temporal um tanto elevado. Por sua vez, a utilização da extensão *WebVisu*, e o acesso remoto via *browser* que daí resultou, constituiu prova de como as tecnologias *Web* podem e devem ser aplicadas no meio industrial de modo a auxiliar engenheiros de manutenção e de supervisão no desempenho das suas funções. Ainda que os resultados obtidos não tenham sido ótimos, reforça-se a necessidade de um trabalho de investigação contínuo, dado o grande potencial desta tecnologia que é o *Codesys*.

4.6 Síntese

O presente capítulo descreveu uma solução tecnológica para visualização de dados de tempo-real baseada em *Codesys*. Para isso foram utilizados o *SoftPLC CODESYS Control Win V3* e a *SoftHMI CODESYS HMI*.

Inicialmente, efetuou-se a contextualização de uma aplicação de paletização de caixas e a respetiva especificação das interfaces a desenvolver.

Numa fase seguinte definiu-se o modelo de controlo a impor aos paletizadores e explorou-se o *software Codesys*, onde tendo o *SoftPLC* como a *SoftHMI* foram programados.

Decididos os cenários a ensaiar e definidos todos os requisitos de interação a cumprir, seguiu-se a programação dos controladores e das interfaces. A esta fase seguiu-se o teste da utilização de *browsers* para aceder às páginas *Web* geradas no servidor do *Codesys*.

Finalizado o trabalho a que nos propusemos, foram expostos e discutidos os resultados obtidos, tendo o *Codesys*, apesar do seu elevado potencial, demonstrado alguma instabilidade a cumprir todos os requisitos impostos.

5 DESENVOLVIMENTO DE SOLUÇÕES DE VISUALIZAÇÃO EM WINCC

Este capítulo descreve o segundo trabalho experimental desenvolvido nesta dissertação. O seu propósito é o desenvolvimento de meios que permitam avaliar as potencialidades de *hardware Siemens* no âmbito da visualização de dados industriais de tempo-real. Para isso, é utilizado um PLC S7-1200 e um painel HMI *SIMATIC KTP600*. O ambiente *TIA Portal* serve a programação dos equipamentos e as *Apps* disponibilizadas pela *Siemens* permitem o acesso remoto à informação fabril de tempo-real. O sistema alvo é diferente do considerado para a solução *Codesys*, mas continua a ser emulado em *Factory I/O*.

Numa fase inicial são expostas algumas das caraterísticas mais relevantes do *hardware Siemens* utilizado, salientando como algumas dessas caraterísticas potenciam a visualização de dados em tempo-real. De seguida é definido e justificado o cenário a implementar no *software* de simulação *Factory I/O*, sendo feita uma breve descrição do problema a tratar e dos respetivos requisitos de controlo e visualização de dados.

Numa fase seguinte são especificadas as interfaces para os diferentes intervenientes (operador, engenheiros de manutenção e supervisão), assim como é introduzido o *software* de desenvolvimento *TIA Portal*. Neste ponto é apresentada não só a modelação do controlo do encaminhador, como também as funcionalidades disponibilizadas pelo *WinCC* para o desenvolvimento das visualizações.

A estes pontos segue-se naturalmente o desenvolvimento e teste das interfaces especificadas. Por fim, faz-se a discussão dos resultados obtidos e a comparação dos mesmos com os encontrados no capítulo anterior.

5.1 Tecnologia *Siemens*

Neste ponto expõem-se as caraterísticas do *hardware Siemens* escolhido. Em primeiro lugar, o painel embutido, que apesar de não representar o equipamento mais atual disponibilizado pela *Siemens*, permite demonstrar que não é necessária a tecnologia mais recente para possibilitar a visualização de dados em tempo-real. Em segundo lugar, o PLC,

compatível com o painel escolhido e adequado para a demonstração de funcionalidade que o presente trabalho pretende comprovar.

Para além destas, salientam-se as opções existentes até à data de *Apps* desenvolvidas pela *Siemens*, que em conjunto com o *hardware* da marca, permitem o acesso em tempo-real a dados da fábrica, com recurso a *tablets* e *smartphones*.

5.1.1 Painel embutido

Para o trabalho em questão optou-se por um painel de gama básica de 1ª geração, representado na figura 5.1; a HMI *SIMATIC KTP600 Basic Color PN*.

Este painel é compatível com a gama de controladores *S7-1200/1500* e *S7-200/300/400*, comunicando facilmente através de redes *PROFINET*. Para além disto, é compatível com controladores de outras marcas como a *Allen Bradley* e *Modicon*, utilizando protocolos mais genéricos como *Ethernet/IP* e *Modbus TCP/IP*, respetivamente [51].

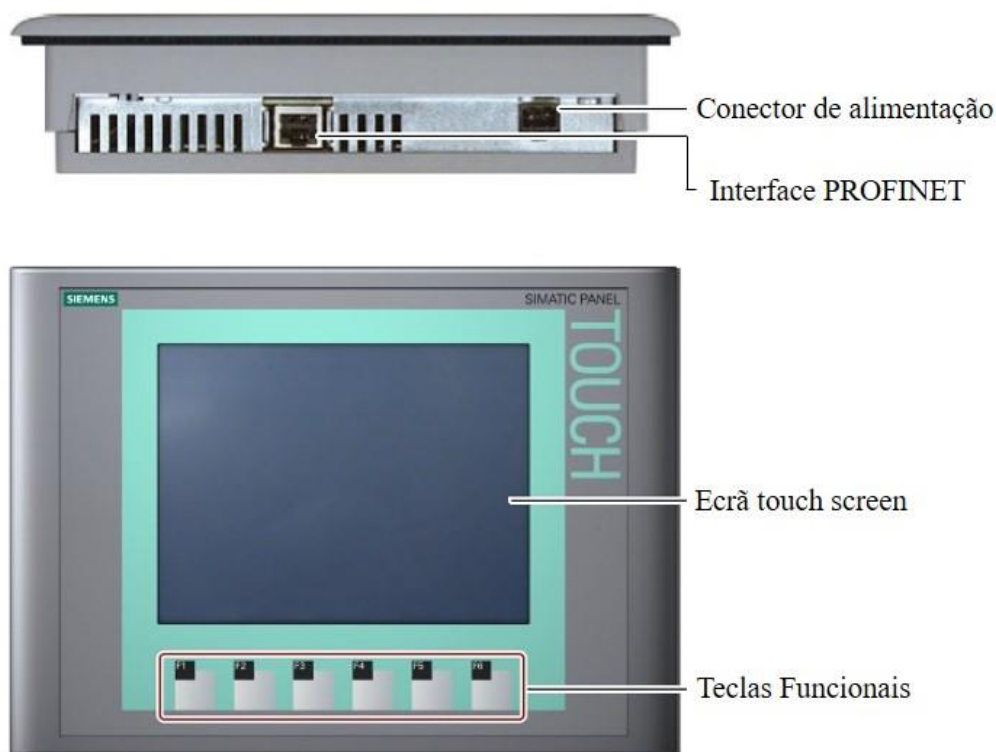


Figura 5.1 – HMI *SIMATIC KTP600 Basic Color PN* [52].

O *software* de configuração do dispositivo é o *WinCC Basic*, integrado no *TIA Portal*, que confere a esta HMI uma variedade de funcionalidades, mediante os objetos que podem ser integrados. As tabelas 5.1 e 5.2 expõem algumas características funcionais da HMI *SIMATIC KTP600* e os objetos que podem ser usados em aplicações de visualização.

Tabela 5.1 – Características funcionais da HMI SIMATIC KTP600 Basic Color PN [51].

Ecrã	TFT <i>Touch screen</i> 5.7 polegadas 256 cores Resolução 320x240 <i>Pixels</i>
Elementos de Controlo	6 teclas funcionais configuráveis Teclados numérico e alfanumérico no ecrã
Alimentação	24V DC
Processador	RISC 32-bit
Memória	1 Mbyte
Interface	Porta <i>Ethernet</i> RJ45

Tabela 5.2 – Funcionalidade da HMI SIMATIC KTP600 sob configuração no WinCC [51].

Integração de objetos numa aplicação para painel básico	
Objetos de Imagem complexos	Objetos de Imagem
Visualização de Alarmes	Campos de texto
Visualização de Tendências (<i>Trend views</i>)	Campos de I/O
Gestão de utilizadores	Campos de data/tempo
Visualização de receitas	Botões e interruptores
Gráficos de barras	Objetos geométricos

Apesar de este painel não possuir a funcionalidade de servidor *Web*, em particular o *Sm@rtServer* da *Siemens*, que versões mais recentes deste equipamento possuem, é possível colmatar esta lacuna selecionando um controlador que inclua um servidor *Web*, como se discute de seguida.

5.1.2 PLC

Como opção de controlo escolheu-se o PLC *Siemens* da série S7-1200 CPU 1214C, uma vez que se trata de um controlador versátil, que permite expansão por blocos modulares de comunicação e de sinal (figura 5.2). O *software* de programação utilizado foi o *STEP 7*, também integrante no *TIA Portal*, e que permite a utilização de três linguagens de programação diferentes: LAD (*LADder logic*), FBD (*Function Block Diagram*) e SCL (*Structured Control Language*). Em termos de comunicação, o módulo CPU escolhido possui uma porta PROFINET integrada (figura 5.2), que é o suficiente para estabelecer a comunicação com a HMI referida anteriormente. No entanto, as capacidades de comunicação deste controlador podem ser expandidas por adição de diversos módulos de comunicação [53].

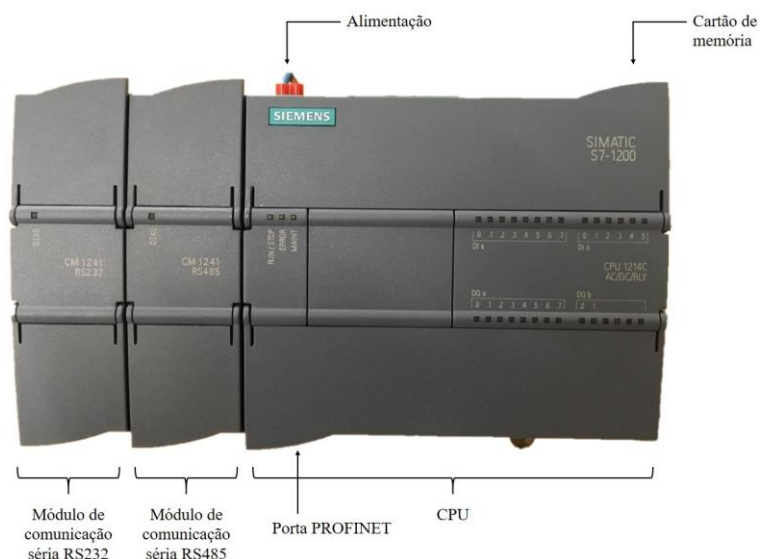


Figura 5.2 – PLC SIMATIC S7-1200, CPU 1214C AC/DC/RLY, com 2 módulos de comunicação série.

Na tabela 5.3 encontram-se algumas características funcionais deste controlador.

Tabela 5.3 – Características funcionais do PLC S7-1200 CPU 1214C AC/DC/RLY [53].

Entradas / Saídas	14 Entradas digitais 2 Entradas analógicas 10 Saídas digitais
Alimentação	300 mA a 120 V AC até 150 mA a 240 V AC
Porta comunicação	Ethernet RJ45
Expansão por módulo de comunicação	Até 3 módulos
Expansão por módulo de sinal	Até 8 módulos

A funcionalidade de particular interesse deste PLC para a aplicação em causa, e cuja aplicabilidade é comprovada em [54], é o servidor *Web* que disponibiliza acesso a dados de sistema e do processo. Esta funcionalidade colmata as limitações do painel escolhido (referidas no final do ponto anterior) facilitando o acesso a dados em tempo-real por meio de um PC ou de dispositivos móveis, sendo as páginas disponibilizadas responsivas à resolução do ecrã do dispositivo utilizado. O S7-1200 suporta tanto páginas *Web standard*, integradas no seu CPU, como páginas *Web* definidas pelo utilizador, desenvolvidas em HTML (*Hyper Text Markup Language*) e recorrendo a comandos AWP (*Automation Web Programming*) pré-definidos para aceder aos dados no CPU [53].

5.1.3 Clientes “App”






Uma pesquisa genérica sobre *mobile Apps* para aplicação industrial, quer na *App Store*, quer na *Google Play*, revela o quão promissora esta solução pode vir a ser para a mobilidade e acessibilidade da informação fabril, dada a forte presença de aplicações

associadas a produtos de fabricantes como a *Rockwell Automation*, a *Omron*, a *Schneider Electric* e a *Siemens*.

A *Siemens* oferece de momento quatro *Apps* que permitem aceder remotamente aos diferentes PLCs/HMIs/SCADAs que comercializa, facilitando a monitorização e controlo dos equipamentos por eles controlados. No entanto, apenas duas delas têm interesse para a aplicação em questão: a *SIMATIC S7* e a *SIMATIC WinCC Sm@rtClient* (tabela 5.4). As duas *Apps* não referidas estão vocacionadas para aceder à série *LOGOSoft* e a sistemas SCADA da *Siemens*, respetivamente.

A *SIMATIC S7* permite aceder ao servidor *Web* integrado no controlador, estando na tabela 5.4 representadas as principais funções disponíveis na aplicação. Já a *SIMATIC WinCC Sm@rtClient* é uma aplicação direcionada para o acesso ao *Sm@rtServer* disponibilizado na 2ª geração de HMIs da *Siemens*, que permite o acesso aos ecrãs da própria HMI remotamente [55, 56].

Tabela 5.4 – Principais funcionalidades das aplicações *SIMATIC S7* [55] e *SIMATIC WinCC Sm@rtClient* [56]

	Compatibilidade	Funcionalidades Principais
Simatic S7  	Controladores: S7-1500, S7-1200, ET 200SP	<ul style="list-style-type: none"> • Acesso a múltiplas estações (até 50 CPUs) para diagnóstico e controlo; • Colocar o CPU dos controladores em modo de corrida ou paragem; • Monitorizar e modificar variáveis ou tags, adicionando-as à lista de tags; • Visualizar variáveis em gráficos de tendências; • Importação de variáveis via Excel; • Controlo de entradas por botões (até 4 lado a lado); • Controlo de saídas por campo de texto (até 4 lado a lado); • Manual do utilizador da App integrado.
Simatic WinCC Sm@rtClient   	SIMATIC HMI: Basic Panels, Comfort Panels, Mobile Panels, SIMATIC WinCC Runtime Advanced com Sm@rtServer integrado	<ul style="list-style-type: none"> • Operação e monitorização móvel e remota de sistemas HMI sobre IWLAN; • Configuração em modo de “monitorização apenas” ou em modo de “operação coordenada”, apenas um utilizador de cada vez tem direito de operação; • Detecção automática de painéis para fácil configuração; • Disponibiliza todo o layout do painel, incluindo botões funcionais físicos.

5.2 Aplicação de suporte e teste

Tal como no capítulo anterior, utilizou-se o *Factory I/O* para desenvolver o alvo de controlo e monitorização. Para o presente caso de estudo optou-se por uma aplicação de encaminhamento de caixas consoante a dimensão das mesmas, uma vez que, tal como se referiu no capítulo 2, estas aplicações se encontram amplamente disseminadas na indústria. A justificação para esta aplicação ser um pouco mais simples do que a considerada no capítulo

anterior reside unicamente no facto de ser suportada por equipamentos com desempenhos mais modestos do que os que decorrem da utilização de soluções *Codesys* executadas em PCs. Porém, os requisitos de visualização são, como se verá, equivalentes aos definidos para a aplicação considerada no capítulo anterior.

5.2.1 Aplicação de encaminhamento

A aplicação alvo consiste então na deteção do tamanho de caixas que se encontram sobre paletes e no seu posterior encaminhamento para diferentes tapetes de rolos – figura 5.3. O sistema tem como ponto central uma mesa de transferência que permite que caixas surgidas no ponto definido pela seta descendente, a verde, sejam levadas até aos pontos definidos pelas setas ascendentes, a vermelho.

Há três modos de funcionamento:

- Automático, sendo as caixas encaminhadas para a direita, frente e esquerda consoante o seu tamanho, pequeno, médio ou grande, respetivamente;
- Semiautomático, onde o operador seleciona, através de botoneiras, para que tapete (direita, frente ou esquerda) são redirecionadas as caixas; neste modo a deteção do tamanho de caixas não precisa de estar ativa, podendo a lógica de separação ser alterada pelo operador;
- Manual, acionamento individual dos diversos atuadores.

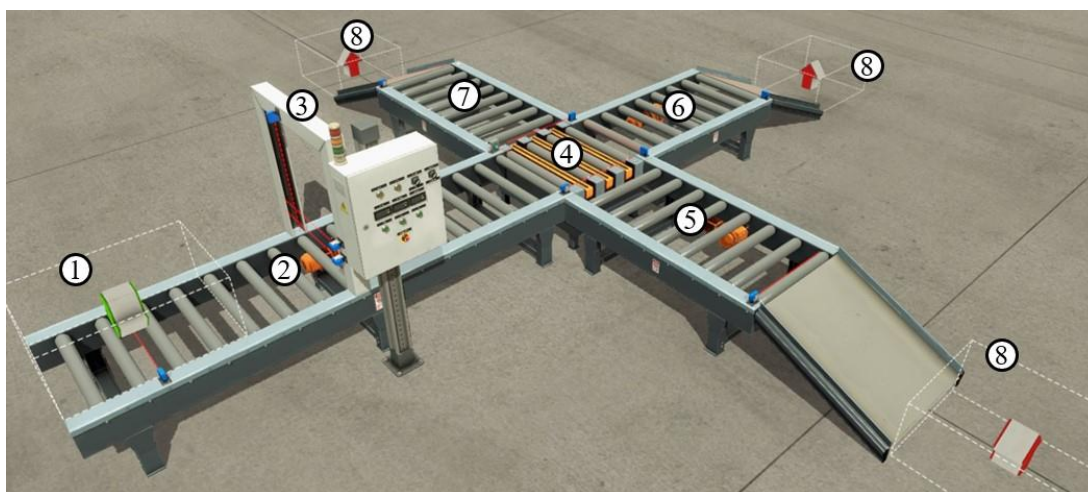


Figura 5.3 – Encaminhador de caixas, consoante a sua dimensão, desenvolvido no Factory I/O.

O encaminhador representado na figura 5.3, é constituído por: (1) emissor de caixas em paletes; (2) tapete de rolos de entrada, com um sensor difusivo à entrada e um sensor capacitivo à saída; (3) três sensores difusivos dispostos de modo a que cada um detete a dimensão da caixa; (4) mesa de transferência por correntes, que permite encaminhar as caixas

para uma das três direções possíveis (direita, frente e esquerda); (5) a (7) tapetes de rolos de saída, cada um deles com dois sensores difusivos, à entrada e saída do tapete, respetivamente; e (8) três elementos eliminadores de caixas, para remoção das mesmas do cenário virtual.

Do cenário virtual fazem também parte botoneiras e sinalizadores luminosos, que permitem a operação do encaminhador no seu modo automático e semiautomático. Nas tabelas 5.5 e 5.6 encontram-se as variáveis de interface entre o controlador e o cenário virtual, sendo que na primeira se encontram as variáveis de controlo do encaminhador e na segunda as variáveis de operação. As variáveis de entrada e saída no controlador consistem, respetivamente, em sinais vindos de sensores e para atuadores na tabela 5.5; e sinais vindos de botões e para sinalizadores na tabela 5.6.

Tabela 5.5 – Variáveis de interface do controlador com o cenário virtual: controlo do encaminhador.

Encaminhador - Tapete de entrada				Encaminhador - Transferência e saída			
Variáveis de entrada (sensores)		Variáveis de saída (atuadores)		Variáveis de entrada (sensores)		Variáveis de saída (atuadores)	
Sensor boxEmitted	Bool	Tapete boxFeeder	Bool	Sensor rightAtEntry	Bool	Mesa tableFront	Bool
Sensor boxAtExit	Bool	boxEmitter	Bool	Sensor rightAtExit	Bool	Mesa tableRight	Bool
Sensor Small	Bool			Sensor frontAtEntry	Bool	Mesa tableLeft	Bool
Sensor Medium	Bool			Sensor frontAtExit	Bool	Tapete exitRight	Bool
Sensor Large	Bool			Sensor leftAtEntry	Bool	Tapete exitFront	Bool
				Sensor leftAtExit	Bool	Tapete exitLeft	Bool
						removerRight	Bool
						removerFront	Bool
						removerLeft	Bool

Tabela 5.6 – Variáveis de interface do controlador com o cenário virtual: operação do encaminhador.

Painel de Controlo Factory I/O			
Variáveis de entrada (botões)		Variáveis de saída (sinalizadores)	
Start	Bool	lightStart	Bool
Stop	Bool	lightStop	Bool
Auto	Bool	warningLight	Bool
SemiAuto	Bool	lightReset	Bool
bReset	Bool	lightRight	Bool
bEmerg	Bool (N)	lightFront	Bool
bRight	Bool	lightLeft	Bool
bFront	Bool	boxesSmall	Word
bLeft	Bool	boxesMedium	Word
		boxesLarge	Word

5.2.2 Cenários de falha a considerar

Uma vez que se trata de uma aplicação mais simples do que a utilizada no capítulo anterior, não existem aqui situações tão críticas no funcionamento do sistema, excetuando um raro encravamento de caixas aquando a transferência para a direita ou esquerda. Por esta razão, de modo a criar uma aplicação que imponha ao utilizador uma necessidade de lidar com situações anómalas, serão introduzidas deliberadamente situações de falha no ambiente virtual. Entre as falhas possíveis de introduzir no *Factory I/O* encontram-se a falha na energização de um atuador, avaria de um sensor ou sensor sempre ativo.

5.3 Especificação de interfaces

Tal como no capítulo anterior, aplicar-se-ão as especificações base, nomeadamente, a exigência de autenticação e a utilização de elementos padrão como menu de navegação e cabeçalho.

Tendo em conta a aplicação descrita, e ainda o *hardware* disponibilizado, a especificação de interfaces neste caso terá em conta as limitações de funcionalidades inerentes a esta escolha.

5.3.1 Especificação da interface do operador

Em termos práticos, a aplicação pressupõe que o painel embutido *KTP600 PN*, está colocado localmente na estação de encaminhamento servindo de interface ao operador.

Em termos funcionais e operacionais, a interface do operador deve ser capaz de:

- ✓ Permitir que o operador selecione o modo de funcionamento do equipamento: manual, semiautomático ou automático;
- ✓ No caso de operação manual, permitir:
 - Dar ordem de iniciar (*Start*), de paragem (*Stop*) e de paragem de emergência (*Emerg*);
 - Acionar os diferentes atuadores do sistema de forma individual;
 - *Feedback* do estado de cada atuador;
- ✓ No caso de operação semiautomática, permitir:
 - Dar ordem de iniciar (*Start*), de paragem (*Stop*) e de paragem de emergência (*Emerg*);
 - Selecionar manualmente o percurso a percorrer por cada caixa;
- ✓ No caso de operação automática, permitir:
 - Dar ordem de iniciar (*Start*), de paragem (*Stop*) em qualquer momento da produção, de purga do equipamento, isto é, esvaziamento da máquina (*Reset*) e de paragem de emergência (*Emerg*);
 - Descartar caixas e ajustar o número de caixas em espera durante a produção, e correspondente posição na lista de espera;
 - Corrigir a dimensão determinada automaticamente durante a produção;
- ✓ Disponibilizar sempre *feedback* do estado de funcionamento do equipamento;

- ✓ Permitir a monitorização da produção: dados atuais sobre o número de caixas que deram entrada, número de caixas de cada tamanho, número de caixas em cada tapete, dados da fila de espera de entrada, etc;
- ✓ Utilização de gráficos para visualização da evolução da produção;
- ✓ Informar o utilizador quanto à existência de situações anómalas de funcionamento de uma forma clara e permitir a aceitação de alarmes.

5.3.2 Especificação da interface do engenheiro de manutenção

Mais uma vez, e com base na solução tecnológica escolhida, a interface de manutenção, será materializada em dois equipamentos, no painel embutido *KTP600* e num dispositivo móvel, munido da *App SIMATIC S7* e/ou de um *browser*.

Em conjunto, estes dois equipamentos devem permitir ao engenheiro de manutenção desempenhar as seguintes funções:

- ✓ Permitir acesso local e remoto;
- ✓ Sinalizar o operador local da sua intervenção no sistema, de modo a que o controlo remoto não leve a situações perigosas para o utilizador local;
- ✓ Permitir assistência remota ao operador, monitorizando variáveis e podendo forçar o valor destas;
- ✓ Permitir monitorizar variáveis ao longo do tempo com recurso a gráficos de tendências;
- ✓ Permitir a monitorização de alarmes de anomalias do equipamento, para uma rápida intervenção do utilizador;
- ✓ Aceder a registos de diagnóstico efetuados pelo PLC;
- ✓ Permitir comando da máquina aquando a instalação da mesma e aquando serviços de manutenção (modo de comando manual);
- ✓ Acesso a manuais do equipamento ou a conteúdos audiovisuais que auxiliem no procedimento de manutenção.

5.3.3 Especificação da interface do engenheiro de supervisão

A interface do engenheiro de supervisão será alicerçada nos mesmos meios que a interface do engenheiro de manutenção, tendo, porém, como objetivo as seguintes funcionalidades:

- ✓ Permitir acesso local e remoto;
- ✓ Sinalizar o operador local do acesso remoto ao sistema;
- ✓ Permitir monitorizar alarmes de sinalização de anomalias de funcionamento do equipamento;
- ✓ Permitir registar evolução de variáveis em gráficos de tendências;
- ✓ Permitir aceder a registos de eventos do funcionamento do equipamento (atuais e históricos);
- ✓ Acesso a dados estatísticos como indicadores de desempenho do equipamento (KPIs): tempo de produção, tempo de manutenção e tempo de paragem.

5.3.4 Árvore de ecrãs

A figura 5.4 representa a árvore de ecrãs a desenvolver para o painel embutido da *Siemens*, que deve corresponder a todas as especificações de operação e a algumas das especificações de manutenção e supervisão. O acesso a esses ecrãs será restringido consoante o nível de autenticação do utilizador. Para além dos ecrãs da figura 5.4, existem janelas *popup* que se sobrepõem a qualquer dos ecrãs referidos de modo a fazer chegar informação prioritária ao utilizador. Estas distinguem-se entre dois tipos:

- Janelas de alarmes ou eventos, nomeadamente: alarmes não confirmados, alarmes pendentes e eventos de sistema ativos;
- Janelas de diálogo (por exemplo, para autenticação).

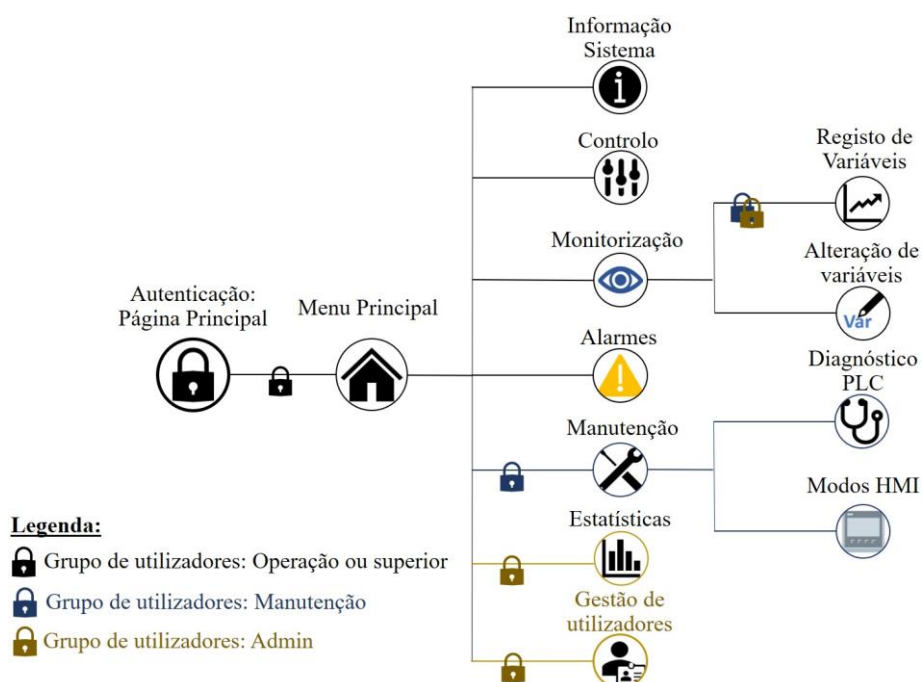


Figura 5.4 – Árvore de ecrãs do painel embutido HMI SIMATIC KTP600.

5.4 TIA Portal: STEP 7 e WinCC

O *TIA Portal* constitui uma das ferramentas mais completas para desenvolvimento e implementação de soluções de automação industrial, estando este *software* segmentado por leque de aplicações. No caso em estudo, é utilizado a versão *Basic V13* deste *software*, que disponibiliza apenas as funcionalidades suportadas pelo *hardware* utilizado. Um dos seus pontos fortes é a modularidade e interoperabilidade entre a programação de aplicações de controlo e de aplicações de visualização, permitindo uma implementação mais rápida de uma solução global. Em termos de linguagens de programação da aplicação e controlo, serão utilizadas duas: LAD e SCL.

Da mesma forma que no capítulo anterior, os modelos de controlo criados para a aplicação de teste encontram-se devidamente explicados no Anexo B.1, não sendo parte fundamental à compreensão do que é descrito neste capítulo.

5.4.1 Detecção de situações anómalas

De modo a alertar o operador da ocorrência de situações anómalas, como avaria de sensores ou atuadores, encravamento de caixas em algum ponto de transferência, ou até inconsistência no número de caixas que deram entrada e saída no sistema, criaram-se alguns blocos capazes de detetar estas situações, tendo em conta o modo de funcionamento do equipamento.

Para a monitorização de sensores, estes blocos comportam-se como *watchdogs*, sendo definidos tempos máximos admitidos para que os sensores estejam ativos ou inativos. Quando esse tempo é ultrapassado, é lançado um sinal de alarme e um código de alarme.

Para a monitorização dos tapetes transportadores e mesa de transferência, divide-se o comportamento dos blocos em três tipos:

- Tipo 1 – bloco para tapete de saída, que monitoriza o sinal do atuador em conjunto com o número de caixas presentes nesse tapete, lançando o sinal de alarme e respetivo código em função da combinação destas duas variáveis;
- Tipo 2 – bloco para tapete de entrada, que para além das duas variáveis do tipo 1 também monitoriza a variável de ocupação da mesa de transferência, lançando o sinal de alarme e respetivo código em função da combinação destas três variáveis;
- Tipo 3 – bloco para mesa de transferência, que monitoriza o sinal do atuador em conjunto com a variável de ocupação da mesa de transferência, lançando o

signal de alarme e respetivo código em função da combinação destas duas variáveis;

Assim, para a monitorização do movimento para a frente da mesa de transferência é utilizado um bloco tipo 3, sendo utilizados para os sinais de transferência para a esquerda e para a direita blocos tipo 1.

5.4.2 Visualizações: objetos e funcionalidades

O WinCC Basic V13 possui uma *toolbox* dividida em três tipos de elementos:

- Objetos básicos (figura 5.5), que não possuem qualquer variável diretamente associada, nem permitem qualquer configuração de eventos, ou seja, não respondem a *inputs* por parte do utilizador, servindo apenas para dar *feedback* quanto ao estado atual de dadas variáveis. Isto é possível configurando tanto animações de visibilidade e/ou aparência, como de movimento;



Figura 5.5 – Objetos básicos do WinCC Basic V13.

- Elementos (figura 5.6), que podem ter ou não variáveis associadas, assim como permitir ou não a configuração de eventos, sendo sempre possível configurar animações de visibilidade, aparência e movimento. Estes objetos constituem assim formas de interação com o utilizador podendo ser utilizados para *inputs* e/ou *outputs* consoante o que for desejado pelo programador;



Figura 5.6 – Elementos do WinCC Basic V13.

- Controlos (figuras 5.7 e 5.8), que constituem objetos complexos, e são capazes de exercer funções pré-definidas no WinCC, sem ser necessária programação adicional de funcionalidade por parte o programador.



Figura 5.7 – Controlos do WinCC Basic V13.



Figura 5.8 – Controlos do WinCC Basic V13, janelas popup e indicador de alarmes.

Outra funcionalidade de interesse é a gestão de utilizadores (figura 5.9), que permite criar grupos de utilizadores, restringindo assim a capacidade de interação com a interface consoante a competência do utilizador. Cada utilizador é atribuído a um grupo, sendo-lhe atribuída uma *password* de acesso e definido se existirá ou não um tempo automático de *logout* após inatividade.

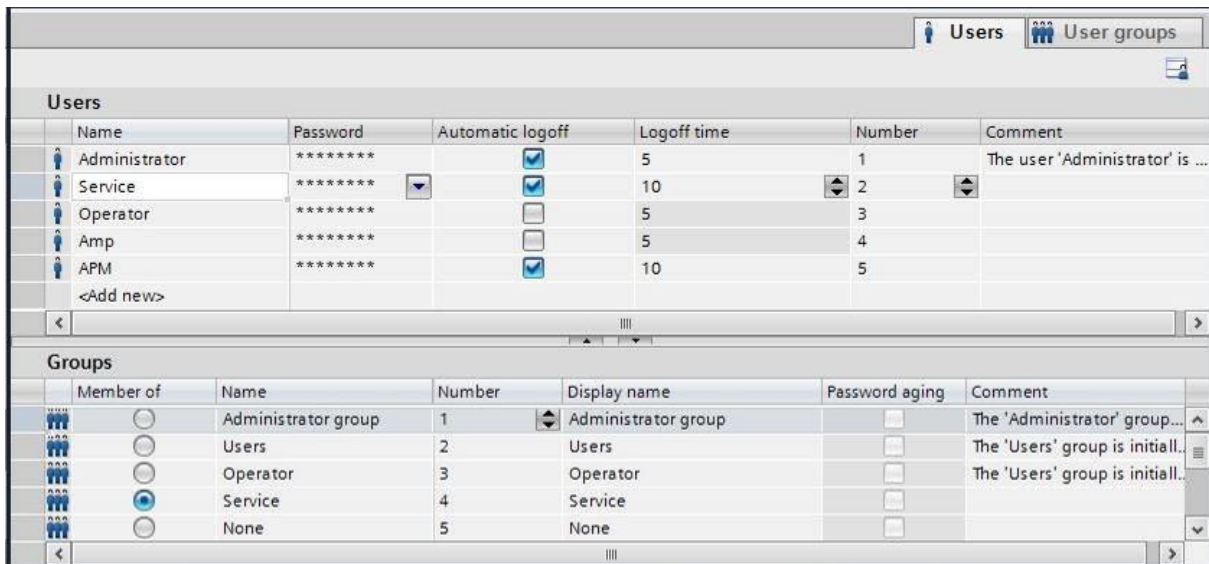


Figura 5.9 – Configuração da gestão de utilizadores em WinCC Basic V13.

Por fim, refere-se como funcionalidade relevante a definição de alarmes (figura 5.10), que podem ser de natureza discreta ou analógica, e atribuídos a uma classe (erro, aviso, sistema, informativo com confirmação ou informativo sem confirmação). Em particular, apenas os alarmes de uma classe que requer confirmação por parte do utilizador podem ser atribuídos a grupos.

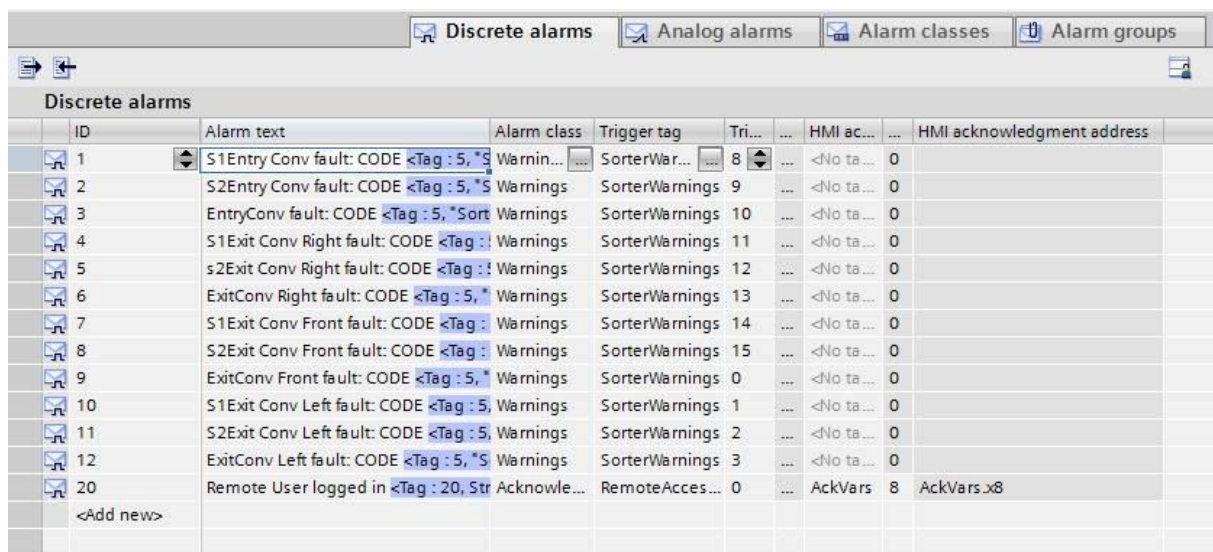


Figura 5.10 – Configuração de alarmes discretos em WinCC Basic V13.

No caso específico de configuração de alarmes discretos, são apenas aceites variáveis do tipo INT ou WORD (variáveis de 16 bits) para ativar um alarme discreto, sendo necessário referir qual o bit dessa variável que está associado ao alarme criado. Para que esse mapeamento esteja correto é preciso ter em conta a forma como as posições dos bits são contadas, o que no caso em estudo ocorre tal como representado na tabela 5.7.

Tabela 5.7 – Contagem de bits nos PLCs SIMATIC S7 e no WinCC Basic V13.

	Byte 0								Byte 1							
	Mais significativa								Menos significativa							
SIMATIC S7 PLCs	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
WinCC	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

5.5 Programação das interfaces

Após a devida configuração de comunicações do equipamento a utilizar e a programação do S7-1200 com o código correspondente aos modelos de controlo acima referidos, deu-se início à programação das interfaces.

No âmbito das especificações base definidas, criou-se um ecrã de autenticação (figura 5.11) tal como no capítulo anterior.

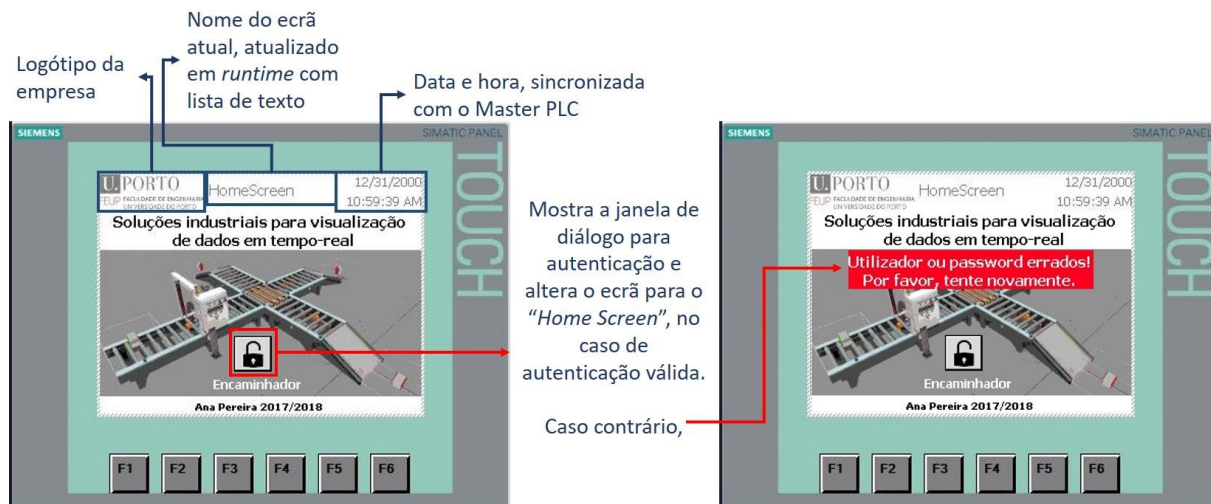


Figura 5.11 – Ecrã de autenticação.

Para os ecrãs criados, excetuando o de autenticação, foi criado um *template* no qual todos eles se baseiam. Nele existem três botões do menu de navegação que estão sempre presentes: o botão *Home*, o botão para acesso à visualização de alarmes e o botão de desligar a aplicação da HMI. Também foram associadas as funções desses botões às teclas funcionais do equipamento F6, F5 e F1, respetivamente.

A figura 5.12 demonstra o ecrã do menu principal, visto por cada um dos níveis de autenticação.

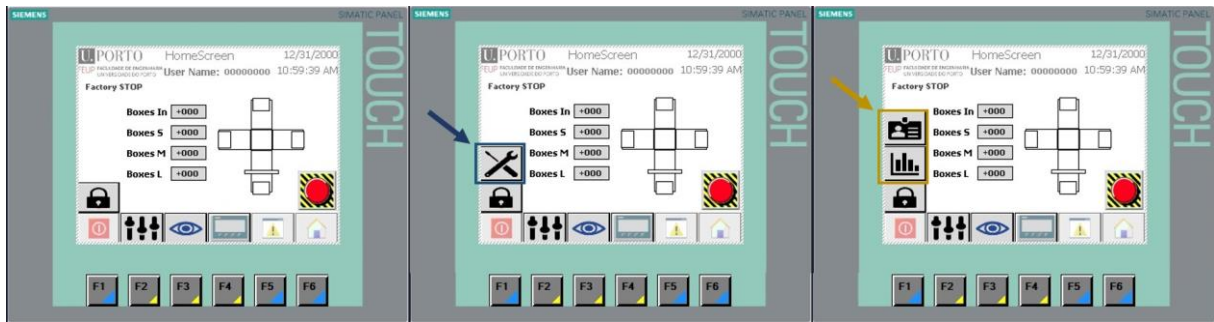


Figura 5.12 – Ecrã do Menu principal (“Home Screen”), visto por um operador (esquerda), um engenheiro de manutenção (centro) e um engenheiro de supervisão (direita).

A figura 5.13 ilustra o ecrã de controlos consoante o modo de operação selecionado: automático, semiautomático ou manual, da esquerda para a direita respetivamente. Quando no ecrã de Controlo, as teclas funcionais F2, F3 e F4 permitem selecionar os modos de operação supramencionados.

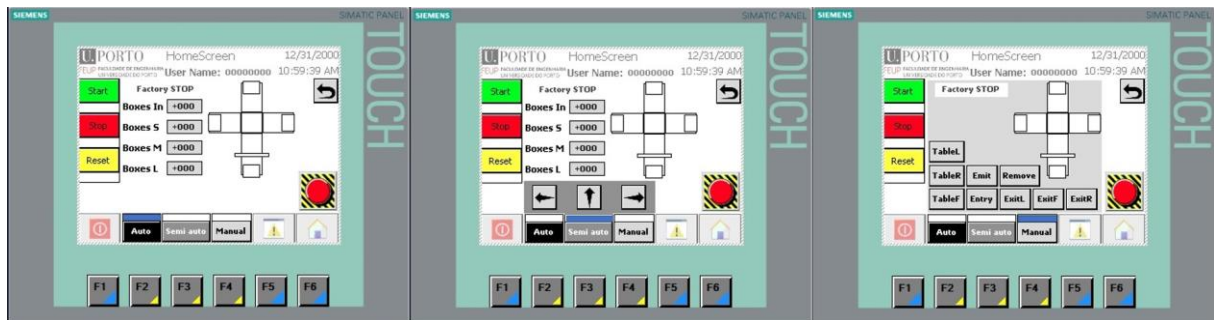


Figura 5.13 – Ecrã de Controlo, com diferentes modos de operação selecionados.

A representação esquemática da linha de encaminhamento foi obtida através dos objetos básicos disponibilizados, e estabelecendo uma animação de aparência e/ou visibilidade associada ao estado de atividade/inatividade de cada elemento.

Já a figura 5.14 ilustra o ecrã de Monitorização. Nele é possível ter um *feedback* mais detalhado quanto ao funcionamento atual do sistema. Também dispõe de dois botões adicionais no menu inferior que permitem o acesso a um ecrã para alteração de variáveis e a outro para o registo de variáveis num gráfico de tendências.

O ecrã de alteração de variáveis serve para executar ações excecionais, como por exemplo, editar a lista de espera no caso de rejeição ou perda de uma caixa; alterar a dimensão de uma caixa, erroneamente detetada; executar *reset* à fila de espera ou contadores de caixas. Por sua vez, o registo de variáveis em gráficos de tendências permite ter uma perceção da evolução dessas variáveis no tempo. De notar que não é possível registar variáveis booleanas desta forma.

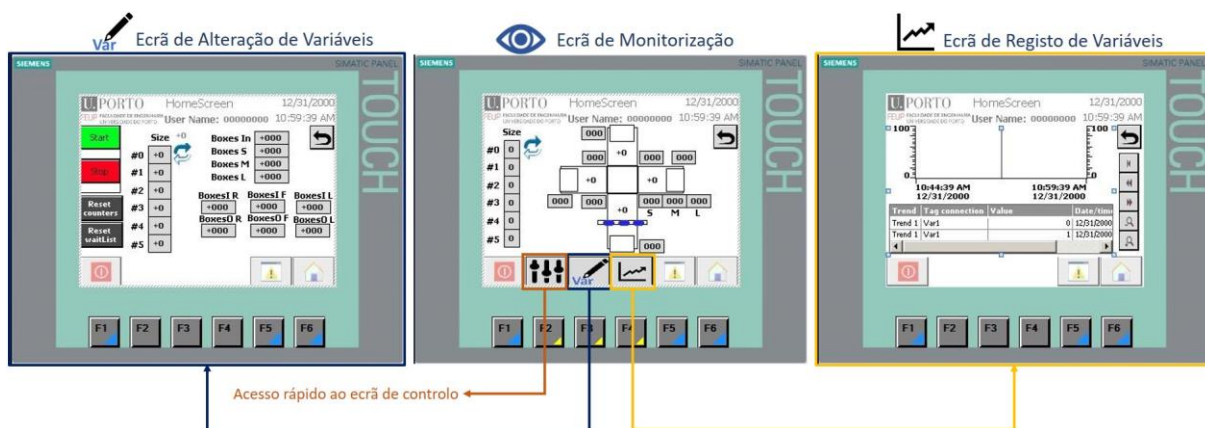


Figura 5.14 – Ecrãs de Monitorização (centro), de Alteração de variáveis (esquerda) e de Registo de variáveis (direita).

Seguem-se os ecrãs de informação do sistema (figura 5.15, esquerda) e de visualização de alarmes pendentes (figura 5.15, direita). Para além do ecrã de visualização de alarmes, o operador é prontamente informado a partir de janelas *popup* (figura 5.15, centro).

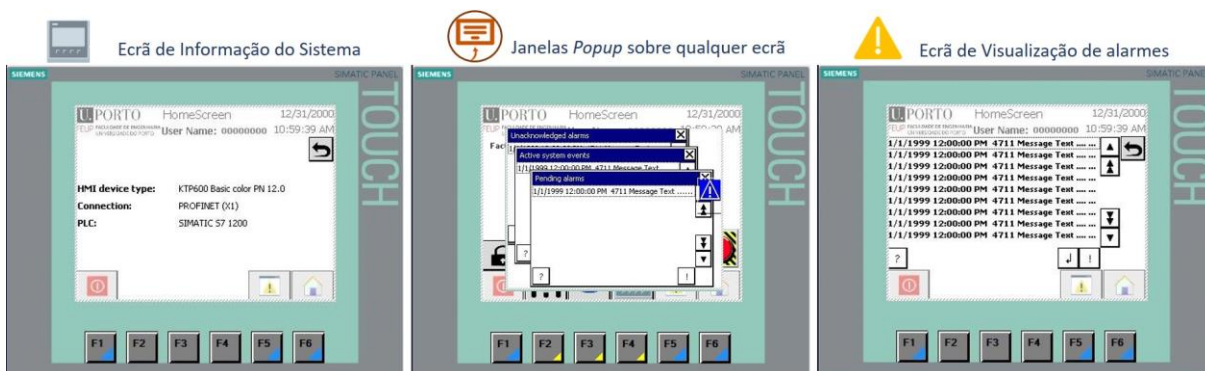


Figura 5.15 – Ecrã de Informação do sistema (esquerda), Janelas Popup (centro) e ecrã de Visualização de alarmes (direita).

A figura 5.16 apresenta alguns dos ecrãs criados para a área de manutenção na interface local. À esquerda encontra-se uma vista geral de todas as monitorizações tanto de atuadores, como de sensores. No mesmo ecrã é possível seleccionar partes do encaminhador (tapete de entrada com sensores, mesa de transferência, etc.) tendo acesso a informação mais detalhada como horas de serviço ou número de ciclos efetuados, número de avarias registadas relacionadas com aquele elemento, ou ainda etapa de funcionamento atual (figura 5.16, centro). Ainda na figura 5.16, à direita, é possível observar o ecrã de diagnósticos do PLC.

Apesar de não estar apresentado na figura 5.16, o ecrã “Modos HMI”, permite alterar o modo de operação da HMI para *online*, *offline* ou transferência.

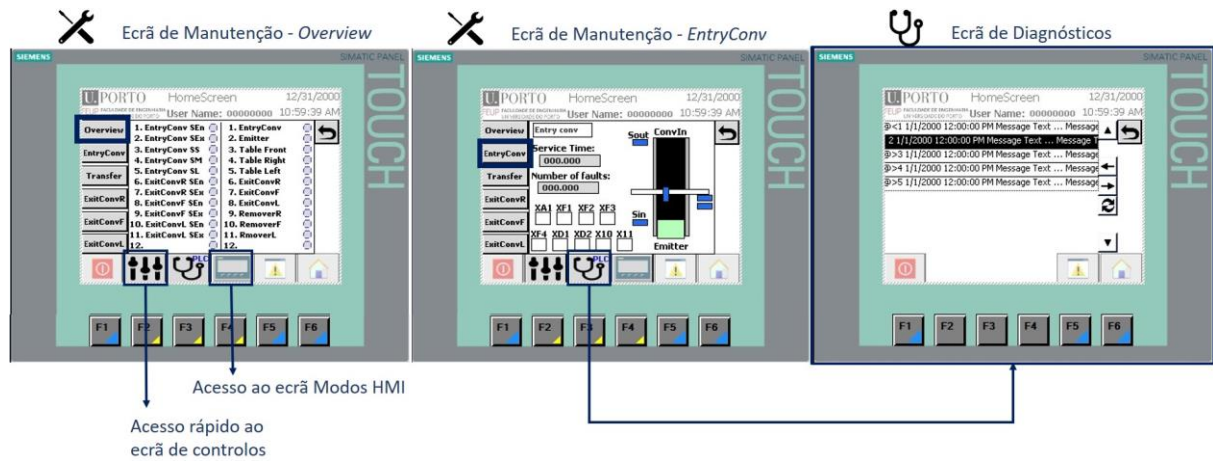


Figura 5.16 – Ecrã de Manutenção – Vista geral (esquerda), ecrã de Manutenção – Tapete Entrada (centro) e ecrã de Diagnósticos de sistema (direita).

Por fim, a figura 5.17 apresenta os ecrãs dedicados a utilizadores de supervisão. Um, à esquerda, que permite a gestão de utilizadores, adicionando/removendo utilizadores ou alterando as suas *passwords* ou níveis de acesso; e outro, à direita, que disponibiliza algumas estatísticas quanto ao funcionamento do encaminhador.

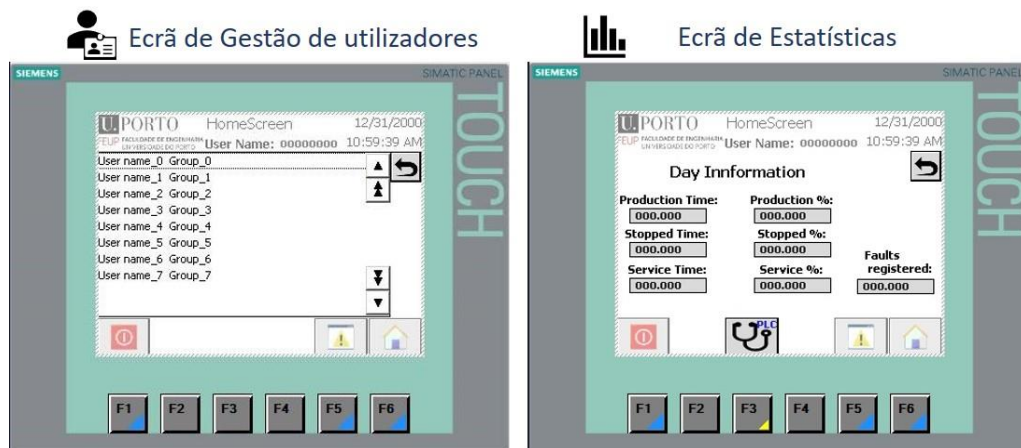


Figura 5.17 – Ecrã de Gestão de utilizadores (esquerda) e ecrã de estatísticas (direita).

Uma vez concebida a árvore de ecrãs com a respetiva funcionalidade para o painel local, convém agora avaliar a solução de acesso remoto para servir os engenheiros de manutenção e supervisão. Para tal, enuncia-se de seguida a configuração necessária ao servidor *Web* do S7-1200.

5.5.1 Configuração do servidor Web do S7-1200

Para ativar o acesso ao servidor *Web* do S7-1200 foi necessário alterar alguns parâmetros na configuração inicial do equipamento, tal como ilustra a figura 5.18. Apesar de não se ter selecionado a aceitação de ligações apenas por HTTPS, salienta-se que em aplicações reais na indústria esta deve sempre estar ativa, de modo a contribuir para a segurança do equipamento. Foi também necessária a criação de uma gestão de utilizadores

dedicada ao acesso ao servidor *Web* do S7-1200, sendo que esta nada tem a haver com a gestão de utilizadores inserida no painel embutido, embora as credenciais possam coincidir.

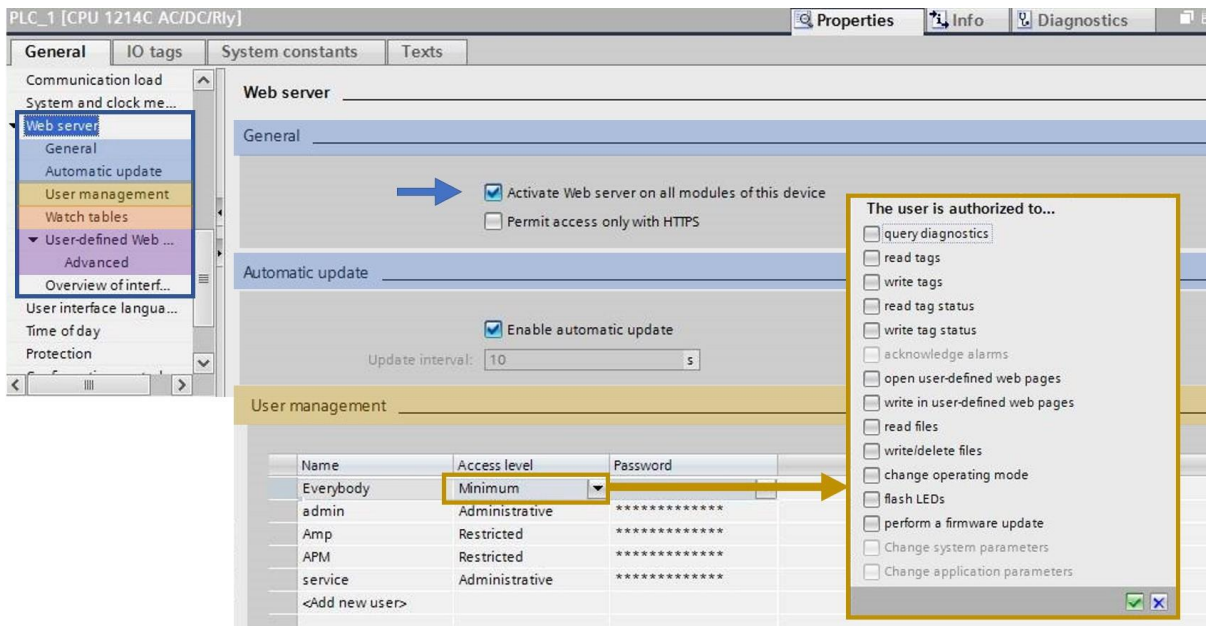


Figura 5.18 – Configuração do servidor Web do PLC S7-1200.

No caso particular das páginas *Web* definidas pelo utilizador, a configuração requer uns passos adicionais tal como mostra a figura 5.19.

De modo a ativar as páginas *Web* definidas pelo utilizador é necessário definir o diretório onde se encontram as páginas do utilizador, assim como definir a página inicial, cujo nome deverá ser “index.html”, e o nome da aplicação, à qual a página é destinada. Uma vez definidos estes parâmetros segue-se a geração dos blocos de dados (DB 333 e DB334 no presente caso) que constituem uma fragmentação dos ficheiros fonte escritos em HTML ou outras linguagens, para que possam ser processados pelo CPU do controlador. A chamada do bloco WWW permite sincronizar as páginas definidas pelo utilizador com o programa.

A figura 5.20 apresenta a página *Web standard* com a lista de *Tags* a serem monitorizadas/modificadas, inseridas pelo utilizador, uma vez que a cada mudança de sessão a tabela elimina as variáveis registadas. Uma opção alternativa seria a utilização de *watch tables* pré-definidas no projeto no *TIA Portal*, uma vez que desempenham a mesma função que a *tag list*, tendo a seu favor o facto de serem memorizadas. Ainda na figura 5.20 pode ser observada uma página *Web* definida pelo utilizador (ou melhor, programador), que no exemplo permite aceder a um documento PDF do manual do utilizador do S7-1200. Isto representa um exemplo muito simples, podendo ser encontradas páginas *Web* definidas pelo utilizador mais avançadas em [57].

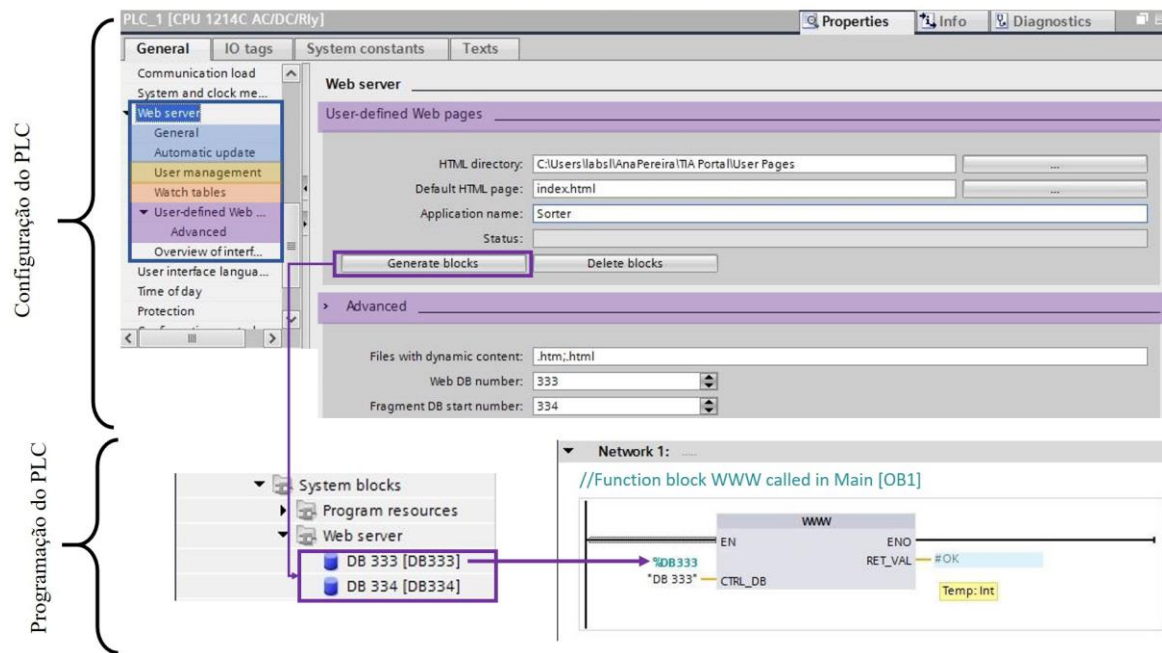


Figura 5.19 – Configuração do servidor Web e correspondente programação para ativar páginas definidas pelo utilizador.

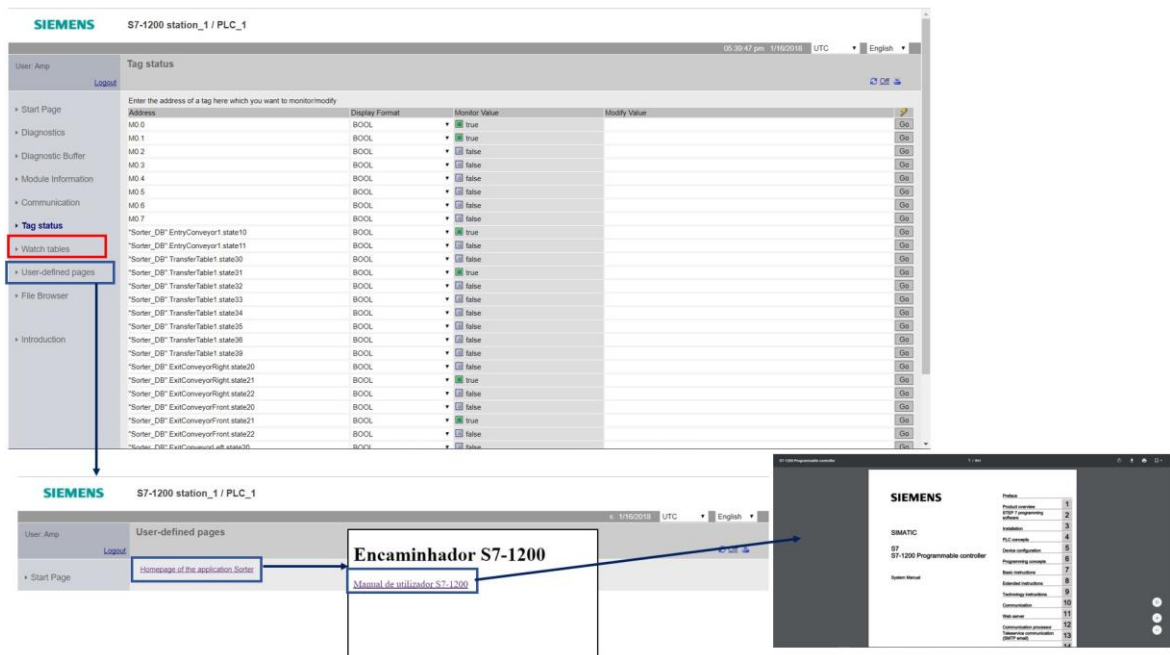


Figura 5.20 – Página Web standard com lista de Tags inserida e página Web definida pelo utilizador, que permite aceder a um ficheiro PDF com o manual do utilizador do PLC.

Para além das páginas Web, é possível aceder ao servidor com a aplicação *SIMATIC S7* da Siemens, como se exemplifica de seguida.

5.5.2 Acesso com a App SIMATIC S7

Uma vez configurado o servidor Web, resta configurar a App *SIMATIC S7* da Siemens. Para tal, basta fazer o *download* da mesma para um dispositivo móvel. Numa primeira utilização da mesma será pedido que atribua uma *password* de acesso à mesma por motivos

de segurança. No caso em questão, por razões práticas esse campo foi deixado em branco. Também é possível no menu de configurações definir a frequência de atualização de dados, sendo o seu valor *default* 0,5s.

A figura 5.21 demonstra a facilidade de configuração e de navegação da App. Para adicionar um CPU basta ir ao menu “*Device Overview*” e adicionar um novo dispositivo, dando-lhe um nome, identificando o endereço IP do dispositivo e ainda introduzindo uma credencial válida, definida aquando da configuração do servidor. Após o dispositivo ser adicionado, o utilizador pode facilmente alterar o modo de funcionamento do PLC, consultar a lista de diagnósticos, obter detalhes sobre *hardware* e ainda aceder rapidamente a dados do CPU, adicionando as variáveis que desejar monitorizar ou controlar à sua “*Tag List*”. As variáveis que constem na “*Tag List*” podem ser visualizadas em gráficos de tendências ou ainda associadas a controlos na “*Control Page*”.

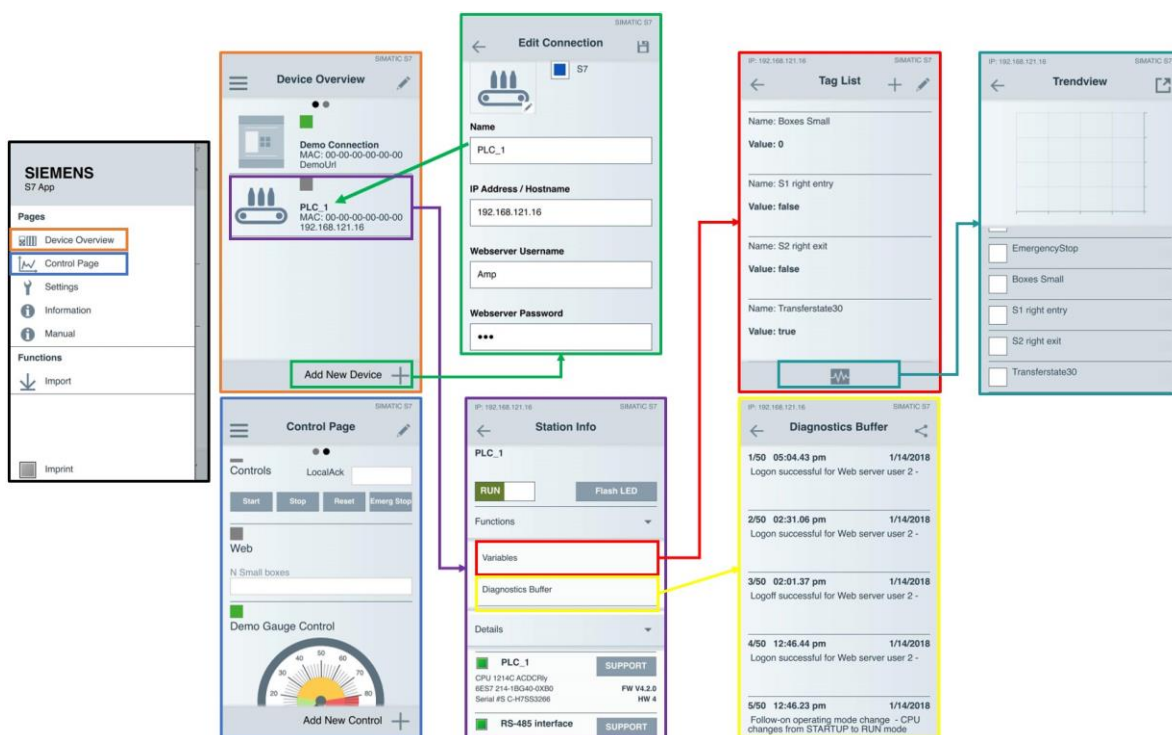


Figura 5.21 – Configuração e navegação da App SIMATIC S7 da Siemens.

Com o intuito de tornar o controlo remoto uma opção minimamente segura para o operador local, estabeleceu-se um “protocolo de *hand shake*” entre o utilizador remoto e o utilizador local. O primeiro, caso pretenda obter direito de operação sobre a estação, deve inserir o seu nome de utilizador na variável “*RemoteUserName*” e colocar a variável “*RemoteUserLogged*” a verdadeira. Esta variável ativará um alarme no painel do operador que terá que ser confirmado ou ignorado consoante o operador pretenda ou não ceder os direitos de operação. Uma vez confirmada a receção do alarme, o utilizador remoto recebe a confirmação na variável “*LocalAcknowledge*”, podendo assim operar a estação. Enquanto

“*RemoteUserLogged*” se mantiver verdadeira, o operador local terá no “*Home Screen*” um indicador quanto ao utilizador remoto (figura 5.22).

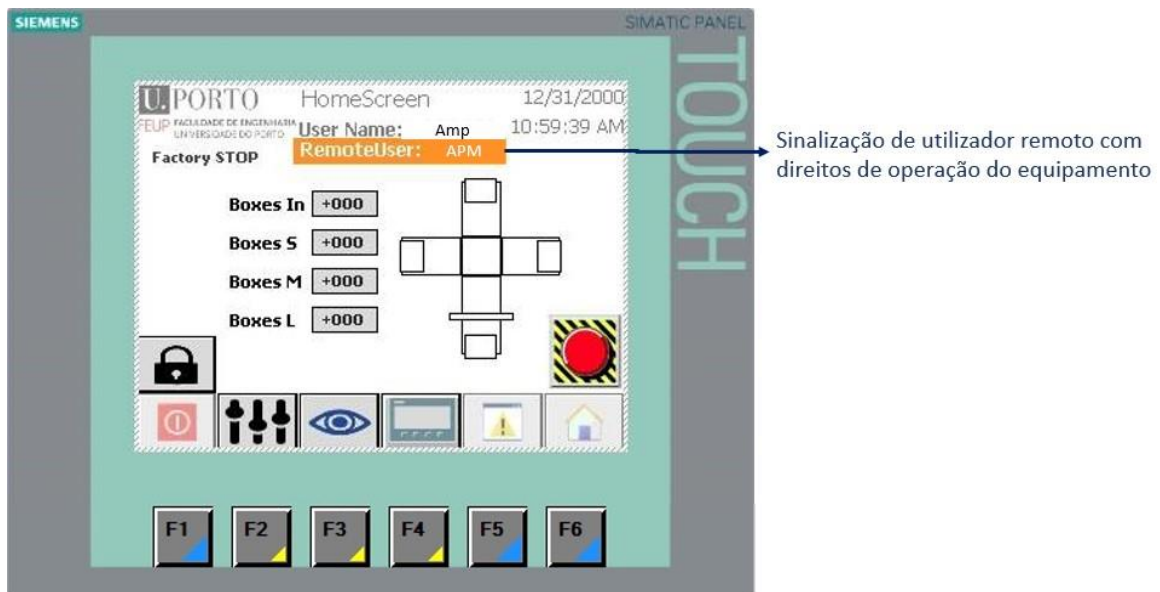


Figura 5.22 – Sinalização de um utilizador remoto com direitos de operação.

5.6 Discussão dos resultados

Do trabalho desenvolvido e exposto neste capítulo, constatou-se que foi possível desenvolver uma interface capaz de responder às especificações descritas no ponto 5.3, tendo sido concretizados os objetivos inicialmente propostos. Foi necessária uma ambientação inicial ao *TIA Portal*, no entanto, após compreensão da estrutura e funcionalidades do programa, foi possível criar a interface desejada com alguma facilidade. Naturalmente, dada a natureza do projeto, nem todas as funcionalidades disponibilizadas pelo *WinCC Basic* foram aplicadas, no entanto, estas devem ser referidas. Entre elas, destaca-se a possibilidade de introduzir gestão de receitas, assim como de utilizar o vasto leque elementos de controlo (interruptores, sinalizadores) disponibilizados nas bibliotecas globais do *WinCC Basic*.

A nível de anomalias introduzidas propositadamente no sistema através do *Factory I/O*, verificou-se uma impossibilidade de injetar avarias em qualquer um dos atuadores, dada a natureza das saídas Q do PLC, que não permitem escrita nesses endereços por parte do *software* de emulação. Por esta razão foram apenas introduzidas falhas nos sensores.

No que diz respeito ao acesso remoto ao servidor *Web* do PLC, verificou-se que um acesso via *browser* com páginas *Web standard* não correspondia a uma solução viável, pelo menos para o S7-1200, que apenas permite uma atualização de dados automática de 10 em 10 segundos, o que não é aceitável para visualização de dados em tempo-real. Também a utilização da funcionalidade “*watch tables*” não foi possível de implementar, uma vez que requeria a utilização do *TIA Portal V14*. Por sua vez, a utilização da aplicação *SIMATIC S7*

revelou-se uma alternativa interessante, dada a facilidade de monitorização de dados diretamente do CPU do controlador, sendo apenas necessário inserir o endereço físico ou simbólico das variáveis para as poder monitorizar. Os únicos inconvenientes associados são a limitação a 50 variáveis por dispositivo e a necessidade de uma pré-configuração da lista de variáveis, uma vez que não é realista assumir que o utilizador saberá “de cor” os endereços das variáveis que quer monitorizar. A utilização da página de controlo da *App* foi testada, dentro dos limites impostos pelo modo de demonstração da mesma, tendo os controlos disponíveis uma usabilidade elevada. O facto de o servidor *Web* permitir múltiplos utilizadores, isto é, utilizadores com credenciais diferentes, facilita que tanto um engenheiro de manutenção, como um engenheiro de supervisão, acessem à informação do seu interesse, sem qualquer interferência entre eles.

Por fim, testou-se a utilização de páginas *Web* desenvolvidas em HTML, ainda que apenas com funcionalidades básicas, e com uma aparência rudimentar, menos apelativa do que a interface da *App* da *Siemens*. Isto permitiu demonstrar a capacidade de aceder a ficheiros PDF, como, por exemplo, manuais do equipamento em questão. No entanto, salienta-se que estas páginas *Web* podem ser melhoradas até atingirem um aspeto gráfico mais apelativo e funcional, implementando gráficos, tabelas e muitas outras funcionalidades típicas de uma HMI tradicional, com recurso a bibliotecas de *JavaScript*. Ainda assim, é de salientar, que quanto maior a complexidade das páginas *Web*, mais será exigido ao processador do S7-1200, o que a certa altura poderá ser prejudicial para o seu desempenho no controlo da aplicação a que está afeto.

De um modo geral, a tecnologia *Siemens* proporcionou ferramentas para um rápido desenvolvimento de interfaces, tendo tanto em vista as necessidades do operador local como as de um utilizador que precise de monitorizar e/ou comandar remotamente através de um *tablet* ou *smartphone*. Ficou assim comprovada a aplicabilidade de *Apps* para tarefas de manutenção e supervisão, o que constituía um dos principais objetivos desta dissertação.

5.7 Balanço das soluções tecnológicas implementadas

Uma vez concluído o projeto, é possível colocar as duas soluções desenvolvidas frente a frente, fazendo assim um balanço dos aspetos positivos e negativos de cada uma.

Ambas as soluções apresentaram diferentes opções para a visualização de dados em tempo-real. A primeira, baseada no acesso remoto via *browser*, demonstrou bastante versatilidade nos objetos gráficos que podiam ser utilizados e visualizados remotamente, no entanto, verificou-se alguma instabilidade nesta solução, com perda de funcionalidade dada a sobrecarga de processamento da aplicação de visualização e também com eventuais falhas de

conexão ao servidor *Web*. Esta última, quer por falha de comunicação, quer por uma atualização accidental da página do *browser*, implicava uma nova autenticação para aceder à interface, o que a certo ponto se tornava inconveniente para o utilizador, introduzindo atrasos nas tarefas que queria desempenhar. O facto de não estar implementado um *design* responsivo, isto é, adaptação do conteúdo da interface ao *display* do dispositivo móvel utilizado, contribuiu para que a atualização accidental da página do *browser* fosse algo recorrente.

Por sua vez, a segunda solução, baseada no acesso remoto via *App*, apesar de limitada em termos de funcionalidades gráficas, demonstrou ser mais estável que a primeira solução, para além de que, por se tratar de uma *App* para dispositivos móveis, vem equipada com um *design* responsivo e funcionalidade *multiTouch*, o que contribui para uma interação com o utilizador mais intuitiva. A tecnologia da *Siemens* pode não ser uma ferramenta fácil de dominar, no entanto, a vasta documentação existente, tanto oficial como proporcionada por outros utilizadores da marca, permite ao engenheiro de automação encontrar respostas rapidamente para qualquer problema com que se depare aquando o desenvolvimento de uma interface gráfica. Este é um dos maiores elementos diferenciadores da tecnologia da *Siemens* face à tecnologia da *3S-Smart Software Solutions*. A primeira disponibiliza ferramentas de trabalho fiáveis e de grande estabilidade, o que já não acontece na segunda.

Ainda assim, deve ser referido, que a *Siemens* é detentora tanto do *software*, como do *hardware* que comercializa, sendo que o mesmo não acontece com a *3S-Smart Software Solutions* e o *software* por eles comercializado, o *Codesys*. Desta forma, é esperado que a *Siemens* possua soluções mais estáveis, uma vez que consegue otimizar o desempenho da sua combinação de equipamento e *software*. Por outro lado, o *software Codesys* pode ser emparelhado com uma grande heterogeneidade de equipamentos, correndo sobre diferentes sistemas operativos, sobre os quais a *3S-Smart Software Solutions* não tem controlo. Isto evidencia que a instabilidade demonstrada pelo *Codesys* ao longo deste trabalho pode também estar associada a alguma incompatibilidade com *hardware* ou sistema operativo utilizados.

A tabela 5.8 apresenta as principais conclusões retiradas deste trabalho de investigação, enunciando a avaliação de cada uma das soluções em diferentes categorias de relevância para o desenvolvimento de soluções para visualização de dados em tempo-real.

Tabela 5.8 – Avaliação das duas soluções encontradas, a primeira em Codesys e a segunda em WinCC.

Categoria de avaliação	Soluções desenvolvidas em Codesys	Soluções desenvolvidas em WinCC Basic
Adaptação ao ambiente de desenvolvimento	●●●●○	●●●○○
Documentação de suporte (oficial e não oficial)	●○○○○	●●●●●
Facilidade de implementação de interfaces	●●○○○	●●●○○
Flexibilidade na programação de interfaces	●●●●○	●●○○○
Funcionalidades básicas de programação de interfaces	●●●●●	●●●●●
Funcionalidades avançadas de programação de interfaces	●●●●●	●●●○○
Comunicação com outros dispositivos	●●●●○	●●●○○
Estabilidade do <i>software</i>	●○○○○	●●●●○
Viabilidade de acesso remoto via <i>Browser</i>	●●●●○	●●○○○
Viabilidade de acesso remoto via <i>App</i>	●○○○○	●●●●○
Implementação de acesso remoto a informação complexa, sem conhecimento de HTML, JavaScript, etc.	●●●●●	●○○○○
Tempo de implementação <u>vs</u> qualidade da solução final	●●○○○	●●●●○
Disseminação no meio industrial	●●●●○	●●●●●

Legenda: ●○○○○ Muito Fraco ●●○○○ Fraco ●●●○○ Razoável ●●●●○ Bom ●●●●● Muito Bom

5.8 Síntese

No presente capítulo produziu-se uma solução tecnológica alternativa à explorada no capítulo anterior, tendo sido para isso utilizados um painel embutido, *SIMATIC KTP600 Basic Color PN*, e um PLC da série S7-1200 da *Siemens*. Nele foram expostas as características mais relevantes do equipamento utilizado, salientando a importância do servidor *Web* para os objetivos a atingir. Seguiu-se a contextualização de uma aplicação de encaminhamento de caixas e a respetiva especificação das interfaces a desenvolver.

Numa fase seguinte definiu-se o modelo de controlo a impor ao encaminhador e explorou-se o *software TIA Portal*, onde tendo o PLC como o painel embutido foram programados. Uma vez recolhidos todos os elementos que permitissem a utilização do *software*, seguiu-se então o desenvolvimento das interfaces, assim como o teste da utilização de páginas *Web standard*, de páginas *Web* definidas pelo utilizador e da *App SIMATIC S7*.

Uma vez finalizado o trabalho, colocaram-se frente a frente as duas soluções desenvolvidas. Ambas obtiveram resultados promissores no que diz respeito ao acesso remoto a informação dentro de ambientes fabris, tendo a solução da *Siemens* demonstrado mais estabilidade e menos funcionalidade face à solução da *3S-Smart Software Systems*.

6 CONCLUSÕES E PERSPETIVAS DE TRABALHO FUTURO

A presente dissertação colocou em destaque o papel desempenhado pelas modernas interfaces gráficas na interação Homem-máquina, demonstrando o quão fulcrais estes equipamentos são para o quotidiano de uma instalação fabril. Para além disso, demonstrou de forma original as mais-valias da implementação de tecnologias *Web* de modo a favorecer a visualização de dados em tempo-real, proporcionando assim uma resposta para as principais necessidades de informação de profissionais das áreas de manutenção e de supervisão.

Para tal, e de modo a contornar a inexistência de um ambiente industrial real, recorreu-se à utilização de ambientes industriais virtuais criados no *software* de emulação *Factory I/O*. Isto permitiu comprovar a aplicabilidade de ambientes virtuais como ferramenta de teste para o desenvolvimento de interfaces gráficas, facilitando desta forma a instrução de estudantes de automação industrial na área do desenvolvimento de HMIs, muitas vezes minimizada face a outras áreas da automação.

Do universo tecnológico existente, optou-se pela utilização de duas tecnologias de grande relevância para aplicações na indústria: a primeira da *3S-Smart Software Systems* e a segunda da *Siemens*. Daqui resultaram duas abordagens diferentes à integração de acesso remoto e mobilidade nas soluções HMI tradicionais: a primeira baseada principalmente em tecnologia *Web*, particularmente em *HTML 5* e *JavaScript*, com recurso a *browsers*; e a segunda alicerçada principalmente na utilização de *Apps*, em particular a *App SIMATIC S7* da *Siemens*.

Este trabalho de investigação permitiu à sua autora familiarizar-se com o desenvolvimento de soluções de interação Homem-máquina, reunindo assim experiência de trabalho com o equipamento e *software* utilizados, assim como adquirindo conhecimento nas melhores práticas de conceção de ecrãs, algo que tem muita influência na qualidade das soluções HMI desenvolvidas.

O projeto desenvolvido alcançou os principais objetivos propostos, tendo sido desenvolvidas e experimentadas várias aplicações apoiadas nas tecnologias acima referidas.

Desta experimentação foi possível fazer uma análise crítica dos resultados práticos obtidos, enunciando-se de seguida as conclusões de maior relevância:

Concluiu-se que o processo de desenvolvimento de interfaces gráficas adequadas a diferentes aplicações é um processo iterativo, que requer um grande à vontade com os *softwares* utilizados, sendo a estabilidade dos mesmos um aspeto imprescindível para obter uma interface fiável e de elevada usabilidade.

No âmbito da acessibilidade remota e mobilidade, um dos grandes focos deste trabalho, conclui-se que a utilização de tecnologias *Web* constitui uma solução viável para complementar a HMI tradicional, e para auxiliar profissionais de engenharia nas suas tarefas, quando estas implicam deslocações por toda a fábrica.

Em concreto, e relativamente ao caso de estudo em *Codesys*, refira-se que foi possível demonstrar um vasto leque de possibilidades no acesso via *Web browser* a dados complexos, desde gráficos de tendências, tabelas de alarmes a elementos de controlo. A facilidade de transposição das visualizações criadas no ambiente de desenvolvimento do *Codesys* para visualizações em páginas *Web*, sem qualquer conhecimento de HTML e *JavaScript* por parte do programador, é outro ponto forte encontrado neste caso. No entanto, a instabilidade recorrente do *software*, assim como a dificuldade em compatibilizar a usabilidade associada a equipamentos como *tablets* e *smatphones* com as necessidades de comunicação de dados em tempo-real, constituem alguns dos pontos negativos encontrados.

Por sua vez, e no que diz respeito ao caso de estudo em *TIA Portal* e respetiva *App SIMATIC S7*, conclui-se que, apesar da incapacidade de base da *App* permitir o acesso a dados complexos como tabelas de alarmes, esta permite uma rápida e prática configuração de elementos de controlo na página de controlos, assim como monitorização e alteração do valor de variáveis à escolha do utilizador, e respetiva representação em gráficos de tendências. Para funcionalidades mais avançadas é necessário o conhecimento base de linguagens como HTML, CSS e *JavaScript* para a programação de páginas *Web* definidas pelo utilizador. Ainda assim, e tendo em conta as limitações encontradas nesta solução, a fiabilidade proporcionada por esta *App*, aliada à navegação típica de dispositivos móveis e a satisfação da comunicação de dados em tempo-real, comprovaram o elevado potencial deste tipo de tecnologia para a indústria.

Da comparação entre os dois casos de estudo, resultou um levantamento de vantagens e desvantagens de cada um face a um conjunto de métricas de avaliação estabelecidas.

Por fim, e decorrendo do levantamento tecnológico realizado neste trabalho, é evidente uma tendência generalizada para a adoção de tecnologias *Web* no meio industrial, o

que valida a necessidade da continuação de investigação nesta área, assim como a necessidade de munir os futuros engenheiros de automação com conhecimentos e ferramentas que os tornem capazes de contribuir para a fábrica do futuro.

6.1 Trabalhos futuros

Com vista a construir sobre o trabalho aqui desenvolvido, e de forma a explorar tendências que foram sendo recolhidas ao longo do mesmo, sugerem-se os seguintes desafios:

- Otimização da quantidade de dados a trocar entre *SoftPLC* e *SoftHMI Codesys*, permitindo um melhor desempenho da interface;
- Desenvolvimento de páginas *Web* mais complexas em HTML para o PLC *Siemens S7-1200*, e respetiva análise do impacto das mesmas no desempenho do controlador;
- Exploração das funcionalidades disponibilizadas no *Codesys* e no *WinCC*, que não foram utilizadas no trabalho desenvolvido;
- Utilização de *Apps* generalistas, isto é, não associadas a um produto específico, capazes de implementar protocolos de comunicação como, por exemplo, *Modbus*;
- Aplicação das soluções encontradas a sistemas de automação reais ou a uma combinação de cenários virtuais com sistemas reais, separados fisicamente, invocando a necessidade de monitorização/controlo remoto;
- Implementação de visualização de dados em tempo-real com recurso a envio de e-mails e/ou mensagens SMS, assim como possível controlo através de palavras-chave numa SMS;
- Implementação da troca de dados em tempo-real com uma base de dados na *Cloud* e desenvolvimento de soluções de visualização na *Web* (ex.: *Open Automation Software*);
- Implementação de realidade aumentada no âmbito da produção e manutenção.

Os desafios enunciados constituem assim possíveis temáticas a serem desenvolvidas por futuros estudantes de Automação, desenvolvendo as suas capacidades em áreas inovadoras e de elevado interesse para a indústria.

REFERÊNCIAS

- [1] Carlos, G., *What are Human Machine Interfaces and Why Are They Becoming More Important?*, in *Machine Design Magazine*. September 2015: [Online]. Disponível em: <http://www.machinedesign.com/iot/what-are-human-machine-interfaces-and-why-are-they-becoming-more-important>.
- [2] Katzel, J., *Information systems: The evolution of the HMI*, in *Control Engineering Magazine*. September 2012, CFE Media: [Online]. Disponível em: <http://www.controleng.com/single-article/information-systems-the-evolution-of-the-hmi/d643c1d8644f73884df2c0827cb31f38.html>.
- [3] Hollifield, B.R., et al., *The High Performance HMI Handbook: A Comprehensive Guide to Designing, Implementing and Maintaining Effective HMIs for Industrial Plant Operations*. 2008: PAS.
- [4] Kromenaker, C., *Tracking HMI advances*, in *Control Engineering Magazine*. November 2015, CFE Media: [Online]. Disponível em: <http://www.controleng.com/industry-news/more-news/single-article/tracking-hmi-advances/de9ddea7b5315b375a9fedbe3299f8d0.html>.
- [5] Gadbois, M., *How to deploy smart HMIs as communication hubs*, in *Control Engineering Magazine*. April 2015, CFE Media: [Online]. Disponível em: <http://www.controleng.com/single-article/how-to-deploy-smart-hmis-as-communication-hubs/e8d0e6ac319b310f281b57979126fb8d.html>.
- [6] Gorecky, D., M. Schmitt, and D.M. Loskyll, *Mensch-Maschine-Interaktion im Industrie 4.0-Zeitalter*, in *Handbuch Industrie 4.0 Bd.4: Allgemeine Grundlagen*, B. Vogel-Heuser, T. Bauernhansl, and M.t. Hompel, Editors. 2017, Springer Vieweg. p. 219-236.
- [7] Blank, D. and D.M. Peissner. *HMI-4.0 - Nutzergenerierte Bedienhilfen*. 2017; Disponível em: <http://www.hmi.iao.fraunhofer.de/#A5>.
- [8] Lee, N.A.N., L.E.G. Moctezuma, and J.L.M. Lastra. *Visualization of Information in a Service-Oriented Production Control System*. in *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*. 2013.
- [9] Idoughi, D., M. Kerkar, and C. Kolski, *Towards new web services based supervisory systems in complex industrial organizations: Basic principles and case study*. *Computers in Industry*, 2010. **61**(3): p. 235-249.
- [10] Lee, A.N. and J.L.M. Lastra. *Enhancement of industrial monitoring systems by utilizing context awareness*. in *2013 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA)*. 2013.
- [11] *SmartFactoryKL*. Disponível em: <http://smartfactory.de/?lang=en>.

- [12] Schmitt, M., et al., *Mobile Interaction Technologies in the Factory of the Future*. IFAC Proceedings Volumes, 2013. **46**(15): p. 536-542.
- [13] Schmitt, M., D. Zühlke, and M. Schmitt, *Smartphones und Tablets in der industriellen Produktion*. atp edition, 2013(03): p. 58-65% V 55.
- [14] Pannone, J.J., *How to best design an HMI system*, in *Control Engineering Magazine*. June 2015, CFE Media: [Online]. Disponível em: <http://www.controleng.com/single-article/how-to-best-design-an-hmi-system/04e8c6cfbc4b73882dc65bc4ff42d864.html>.
- [15] PARLAMENTO EUROPEU E DO CONSELHO, *DIRECTIVA 2006/42/CE DO PARLAMENTO EUROPEU E DO CONSELHO de 17 de Maio de 2006 relativa às máquinas*. 2006.
- [16] International Organization for Standardization, *ISO 9241-1:1997 Ergonomic requirements for office work with visual display terminals*. 1997. p. 7.
- [17] International Organization for Standardization, *ISO 11064-1:2000 Ergonomic design of control centres -- Part 1: Principles for the design of control centres*. 2000. p. 30.
- [18] The International Society of Automation, *ANSI/ISA-101.01-2015, Human Machine Interfaces for Process Automation Systems*. 2015. p. 64.
- [19] ASM Consortium, et al., *ASM Consortium Guidelines: Effective Operator Display Design*. 2008: ASM.
- [20] Brandl, D., *Who is building your HMI?*, in *Control Engineering Magazine*. June 2013, CFE Media: [Online]. Disponível em: <http://www.controleng.com/single-article/who-is-building-your-hmi/fc0056d95476eea85ebfa30bb5728dfd.html>.
- [21] Dix, A., et al., *Human-computer interaction*. Third Edition ed. 2004: Pearson Education, Ltd. 834.
- [22] Hollifield, B., *A High Performance HMI: Better Graphics for Operations Effectiveness*, in *ISA Water & Wastewater and Automatic Controls Symposium*. 2012: Holiday Inn Castle Resort, Orlando, Florida, USA. Disponível em: <http://isawwsymposium.com/wp-content/uploads/2012/07/WWAC2012-invited BillHollified HighPerformanceHMIs paper.pdf>.
- [23] Goetz, H., *Part 1: Developing high-performance HMIs*. *Control Engineering Magazine*, 2016. **63**(5): p. P8-P10.
- [24] Goetz, H., *Part 2: Developing high-performance HMIs*. *Control Engineering Magazine*, 2016. **63**(7): p. P1-P6.
- [25] Mital, A. and A. Pennathur, *Perspectives on Designing Human Interfaces for Automated Systems*, in *Handbook Of Industrial Automation*. 2000, CRC Press.
- [26] Errington, J., et al., *Establishing Human Performance Improvements and Economic Benefit for a Human-Centered Operator Interface: An Industrial Evaluation*. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 2005. **49**(23): p. 2036-2040.
- [27] BoschPackagingUsa, *Automated Packaging of Bars*. 2010. Disponível em: <https://www.youtube.com/watch?v=KpkC4GmBsW8>.
- [28] TopTier. *TopTier Palletizers*. 2017; Disponível em: <http://www.toptier.com/m/index.html>.
- [29] Siemens, *Machine-level visualization with SIMATIC HMI: Efficient to a new level*. 2017. Disponível em: <https://c4b.gss.siemens.com/resources/images/articles/dffa-b10135-01-7600.pdf>.

- [30] Omron, *NB-series Programmable Terminals: NB-Designer Operation Manual*. 2017.
- [31] Omron. *Programmable Terminals: NB Series*. 2011; Disponível em: http://industrial.omron.com.br/uploads/arquivos/nb_v412_csm1002572.pdf.
- [32] Harwell, R., *Cover Story: Integrated HMI and PLC: The heart of a "lean automation" solution*, in *InTech Magazine*. July/August 2012, ISA Publications: [Online]. Disponível em: <https://www.isa.org/standards-publications/isa-publications/intech-magazine/2012/august/cover-story-integrated-hmi-plc/>.
- [33] Moeller, *HMI or HMI-PLC – the CompactFlash™ decides: Product Information XV400 Touch Displays*. 2004: [Online]. Disponível em: http://klocknermoeller.com/automation/pdf/289144_W2700-7561GB_HMI_HMI-PLC.pdf.
- [34] UNITRONICS. *UNITRONICS® Samba™ Series*. 2017; Disponível em: <https://unitronicsplc.com/programmable-controllers-samba-series/>.
- [35] Pham, D., *Combined HMI+PLC Unit vs. Separate HMI & PLC System*, in *Automation.com Magazine*. October 2014, ISA: [Online]. Disponível em: <https://www.automation.com/automation-news/article/combined-hmipLC-unit-vs-separate-hmi-plc-system>.
- [36] Siemens. *Mobile Panels - Operator Devices - Siemens*. [Online] 2016; Disponível em: <http://w3.siemens.com/mcms/human-machine-interface/en/operator-devices/advanced-hmi-panel-based/mobile-panels/Pages/Default.aspx>.
- [37] KUKA. *Robot controllers*. 2017; Disponível em: <https://www.kuka.com/en-us/products/robotics-systems/robot-controllers/>.
- [38] Florida, A. *Mobile HMI Panel from Siemens*. 2013; Disponível em: https://www.youtube.com/watch?v=c1KYF_Kb8kI.
- [39] Siemens, *SIMATIC HMI: HMI device: Mobile Panel 277F IWLAN V2, Mobile Panel 277F IWLAN (RFID Tag): Operating Instructions*. 2011.
- [40] 3-S Smart Software Solutions GmbH, *CODESYS® Runtime*. 2014.
- [41] John, K.-H. and M. Tiegelkamp, *IEC 61131-3: Programming Industrial Automation Systems*. 2ª ed. 2010: Springer.
- [42] 3S-Smart Software Solutions GmbH, *CODESYS® Visualization*. 2014.
- [43] 3S-Smart Software Solutions GmbH. *CODESYS Online Help*. 2017; Disponível em: <https://help.codesys.com/>.
- [44] 3S-Smart Software Solutions GmbH, *CODESYS OPC Server V3 - Installation and Usage*.
- [45] 3S-Smart Software Solutions GmbH, *Webinar: CODESYS HMI SL*. 2017.
- [46] w3schools.com. *W3Schools Online Web Tutorials*. 2017; Disponível em: <https://www.w3schools.com/default.asp>.
- [47] tutorialspoint.com. *HTTP Tutorial*. 2017; Disponível em: <https://www.tutorialspoint.com/http/>.
- [48] Terezinho, F., *Automation IT: Remote access, any time, any place - ISA*, in *InTech Magazine*. Septemeber/October 2012: ISA Publications. Disponível em: <https://ww2.isa.org/standards-publications/isa-publications/intech-magazine/2012/october/automation-it-remote-access-any-time-any-place/>.
- [49] Software, O.A. *IIoT, Industrial Software, Data Automation, IOT OPC - Open Automation*. 2017; Disponível em: <https://www.openautomationsoftware.com/>.

- [50] Gadbois, M. and J. Payne, *Industrial controls go mobile*, in *Machine Design Magazine*. January 2014. Disponível em: <http://www.machinedesign.com/technologies/industrial-controls-go-mobile>.
- [51] Siemens, *Data sheet: 6AV6647-0AD11-3AX0*. 2017: Electronic publication. Disponível em: <https://support.industry.siemens.com/cs/ww/en/pv/6AV6647-0AD11-3AX0/td>.
- [52] Siemens, *HMI devices Basic Panels: Operating Instructions*, in *Operating Instructions*. 2012: Electronic Publication. p. 162.
- [53] Siemens, *S7-1200 Programmable controller: System Manual*. 2015. p. 785-860.
- [54] Gavrilov, N.V. *Appliance of WEB-technologies in automation of industrial facilities*. in *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. 2017.
- [55] Siemens. *SIMATIC S7 - The Future of Manufacturing - Siemens*. [Online] 2017; Disponível em: <http://w3.siemens.com/topics/global/en/industry/future-of-manufacturing/industry-apps/s7/Pages/simatic-s7.aspx>.
- [56] Siemens. *SIMATIC WinCC Sm@rtClient - The Future of Manufacturing - Siemens*. [Online] 2017; Disponível em: <http://w3.siemens.com/topics/global/en/industry/future-of-manufacturing/industry-apps/wincc-smartclient/Pages/simatic-wincc-smartclient.aspx>.
- [57] Siemens. *Creating and using user-defined web pages on S7-1200*. 2017; Disponível em: <https://support.industry.siemens.com/cs/document/58862931/creating-and-using-user-defined-web-pages-on-s7-1200?dti=0&lc=en-US>.

(Todas as referências em página Web estavam disponíveis a 22 de janeiro de 2018, data de entrega da versão provisória da dissertação.)

A ANEXO: DETALHES DO DESENVOLVIMENTO DE SOLUÇÕES EM *CODESYS*

Este anexo tem como objetivo explicar de forma mais detalhada o desenvolvimento das soluções de visualização desenvolvidas em *Codesys*.

A secção que se segue expõe os modelos de controlo utilizados na aplicação de paletização, disponível na figura 4.1.

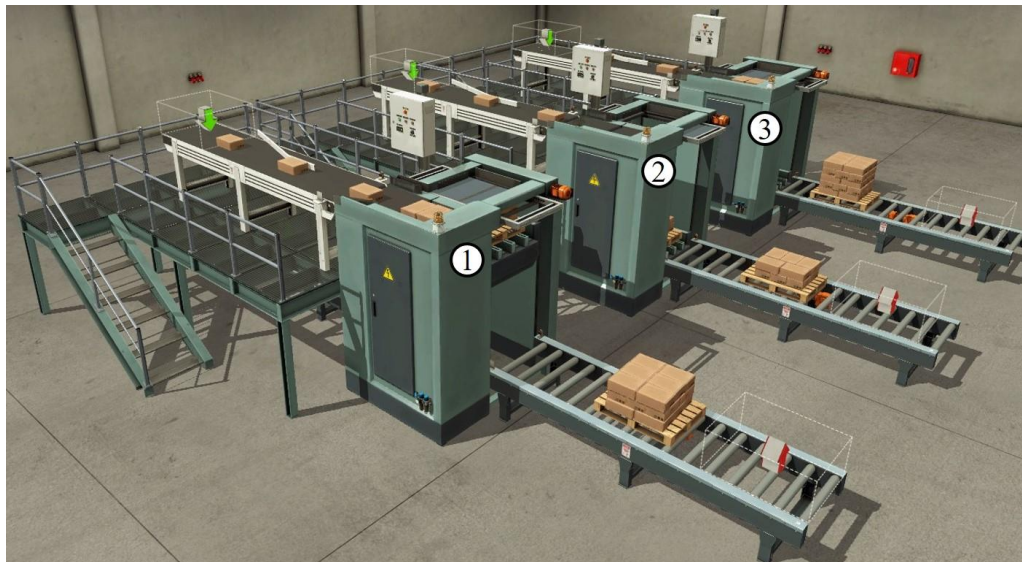


Figura 4.1 – Célula de paletização desenvolvida no Factory I/O.

A.1 MODELOS DE CONTROLO DO PALETIZADOR

Para modelar o controlador de um paletizador utilizou-se a linguagem GRAFCET, que permitiu a estruturação e descrição do comportamento sequencial do equipamento. Optou-se pela utilização de Grafcets hierárquicos para uma melhor perceção do controlo de cada um dos subsistemas acima descritos, assim como dos diferentes modos de operação.

Na figura A.1.1 encontra-se o Grafcet G0 de alto nível que descreve os modos de operação do paletizador, sendo as respetivas etapas descritas em mais detalhe na tabela A.1.1.

Tabela A.1.1 – Descrição das diferentes etapas do Grafcet comportamental de alto nível dos modos de operação e de paragem do paletizador.

Estados Grafcet G0	Descrição
A1	Estado de paragem por inatividade (<i>Standby</i>), permite configuração do equipamento, saídas desativadas.
A2	Estado de paragem devido a ordem de operação automática terminada (Stop no fim do ciclo) , saídas desativadas.
A3	Estado de paragem em qualquer momento de modo de operação automático (Stop), inibe transição de estados nos Grafkets de baixo nível e desativa as saídas.
A4	Estado de paragem (pausa) em qualquer momento de modo de operação automático, desativa os emissores de paletes e de caixas, atinge paragem por falta de alimentação de caixas.
D1	Estado de paragem de emergência, força Grafkets de baixo nível para estados iniciais e desativa as saídas.
D2	Estado de paragem por falha no equipamento, inibe transição de estados nos Grafkets de baixo nível e desativa as saídas.
F1	Estado de corrida em modo de operação automático.
F2	Estado de corrida de inicialização.
F3	Estado de corrida de purga do equipamento, esvazia a máquina.
F4	Estado de corrida em modo de operação manual.

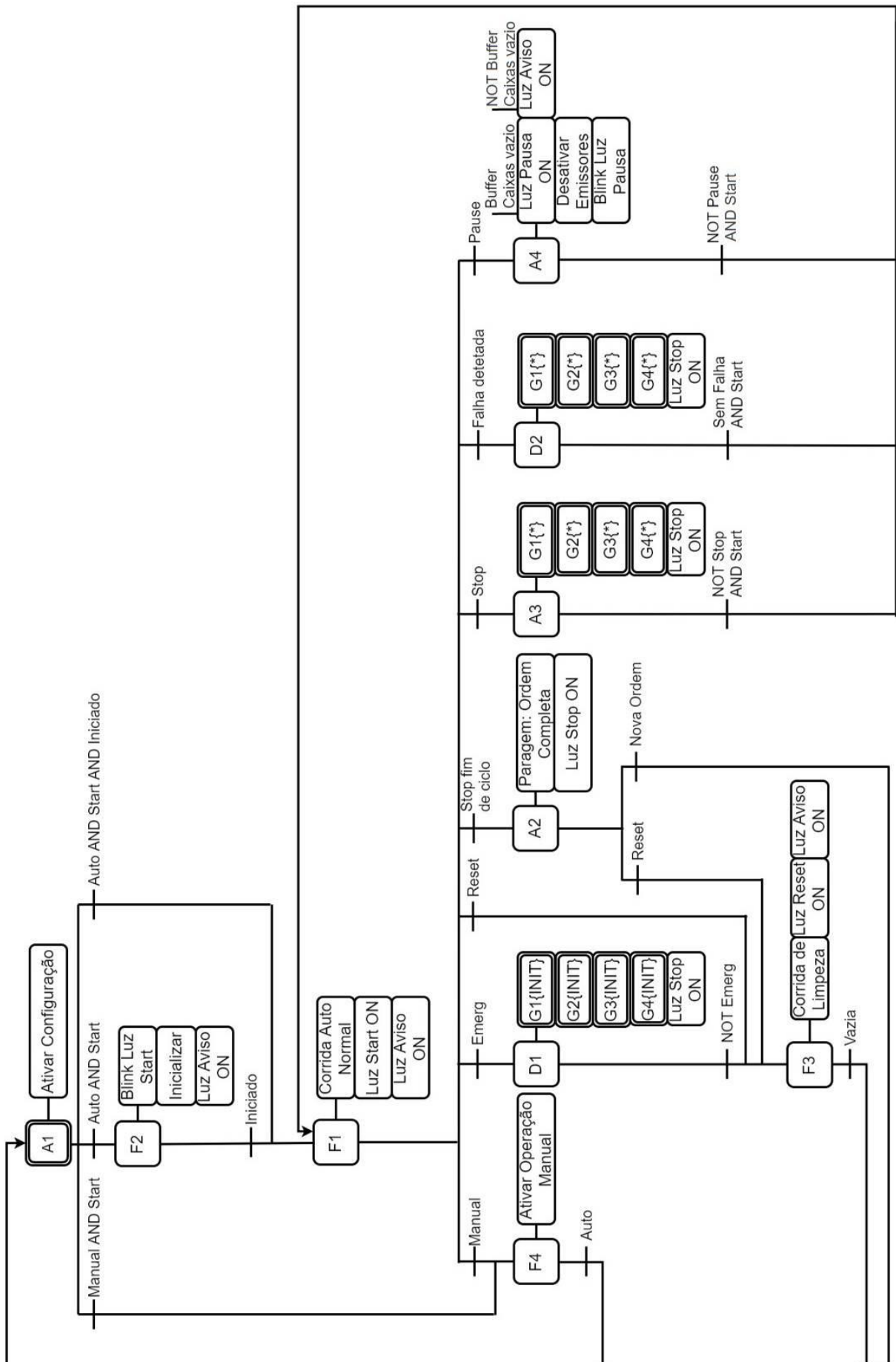


Figura A.1.1 – Grafcet comportamental G0 de alto nível dos modos de operação e de paragem do paletizador.

O Grafcet G0 atua sobre os Grafcets de baixo nível G1, G2, G3 e G4, que descrevem o comportamento de cada um dos subsistemas A, B, C e D do paletizador, respectivamente. Apesar da figura A.1.1 não o representar, para além da etapa F1, todas as etapas que requerem que o sistema esteja ativo, nomeadamente F2, F3, F4 e A4, permitem atuação da emergência, transitando se seguida para a etapa D1.

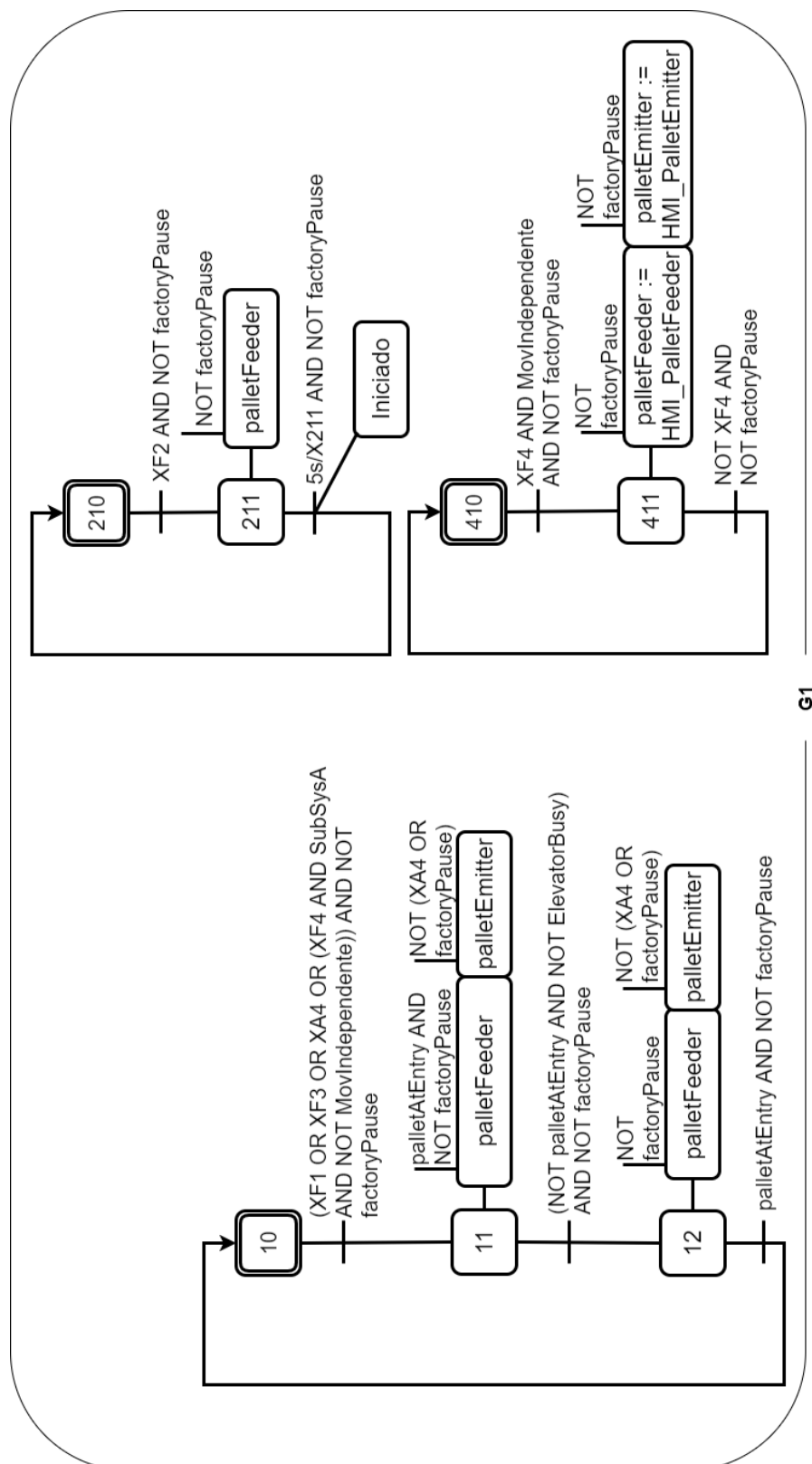


Figura A.1.2 – Grafcet comportamental G1 do alimentador de paletes.

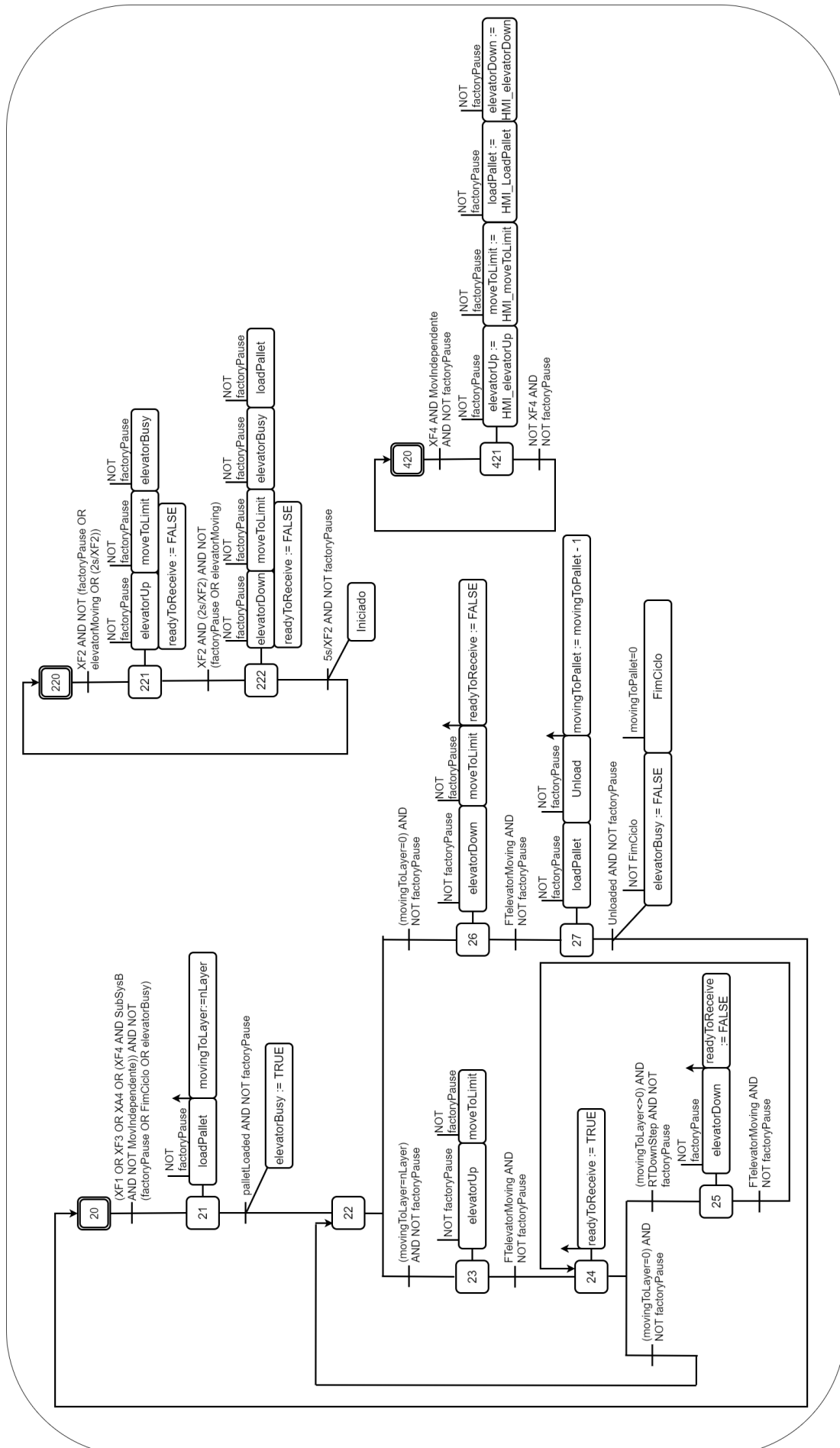
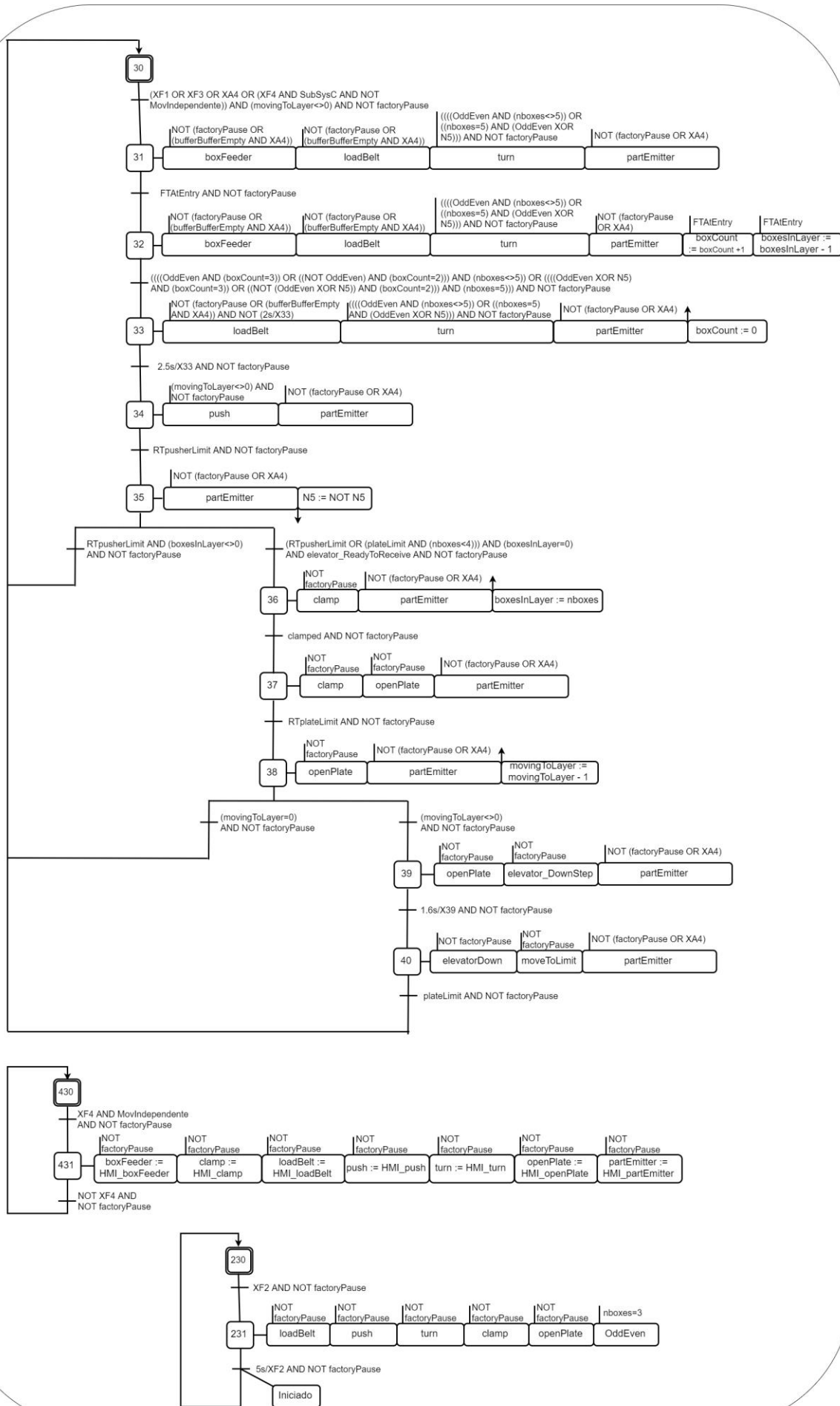
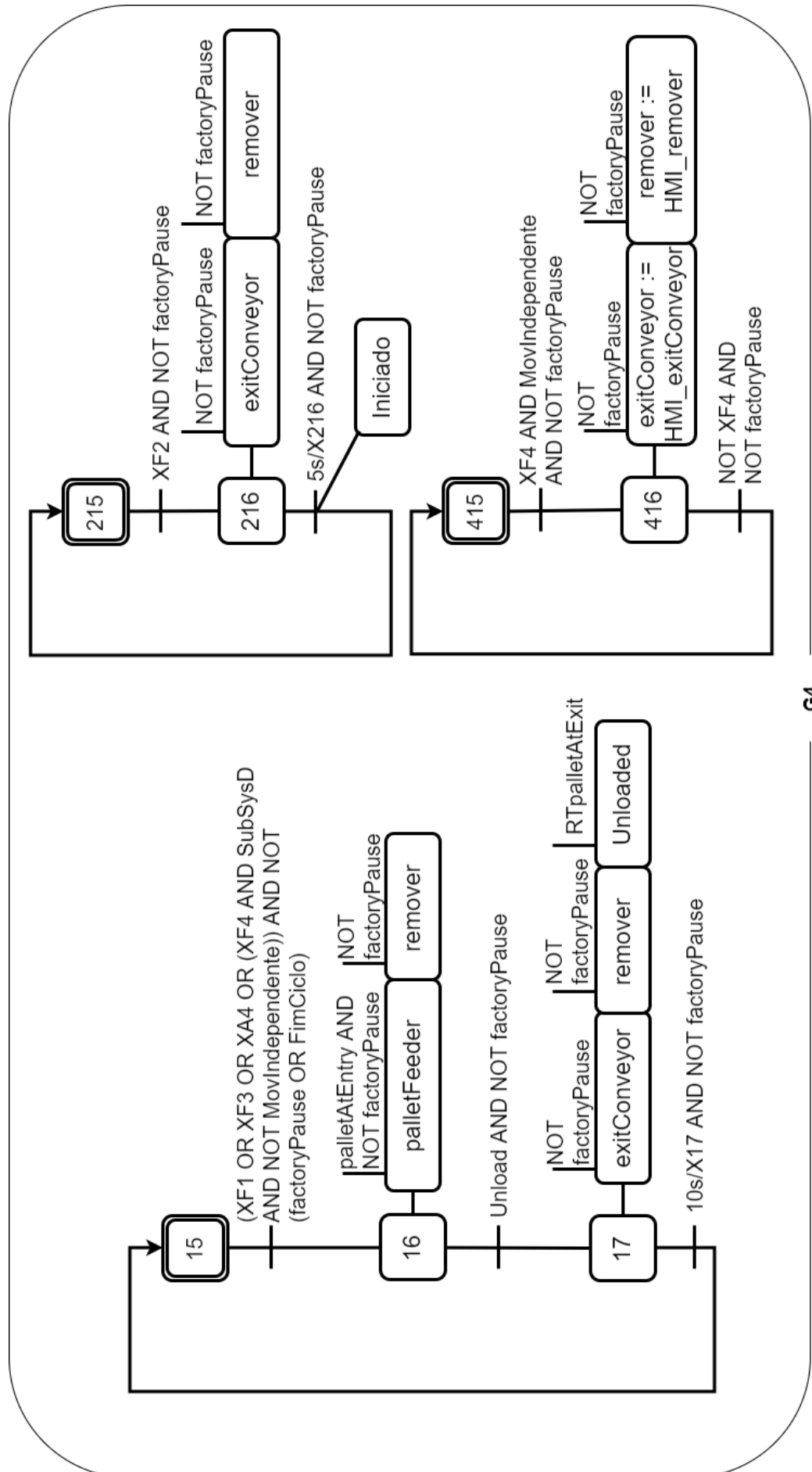


Figura A.1.3 – Grafcet comportamental G2 do elevador de paletes.



G3

Figura A.1.4 – Grafcet comportamental G3 do alimentador de caixas.



G4

Figura A.1.5 – Grafset comportamental G4 do descarregador de paletes.

As figuras A.1.2, A.1.3, A.1.4 e A.1.5 apresentam os modelos comportamentais dos quatro subsistemas do paletizador. Realça-se que as ações forçadas do Grafcet G0 sobre G1, G2, G3 e G4 não estão representadas explicitamente nos Grafcets de baixo nível, isto de modo a simplificar o desenho e compreensão dos mesmos.

Em termos de notação específica, RT* e FT* referem-se a transições ascendentes e descendentes de variáveis booleanas, respetivamente. Também as variáveis X* representam as respetivas etapas de atividade do Grafcet G0.

De modo a garantir que o *SoftPLC* acompanha os estados de funcionamento do cenário virtual no *Factory I/O* (corrida, paragem e reiniciação) implementou-se uma pequena máquina de estados como se observa na figura A.1.6. Deste modo, sempre que o cenário está em corrida, o *SoftPLC* corre também as rotinas de controlo dos três paletizadores; e sempre que o cenário está em paragem ou reinicia, todas as POU's de FBs são reiniciadas, mantendo apenas as variáveis retentivas (caso existam), permitindo que as rotinas de controlo voltem a um estado inicial, tal como ocorre no cenário virtual. A pausa do cenário virtual é o único estado que requer acesso aos Grafcets de baixo nível do paletizador (para inibir transições de etapa), pelo que não se encontra representado na máquina de estados.

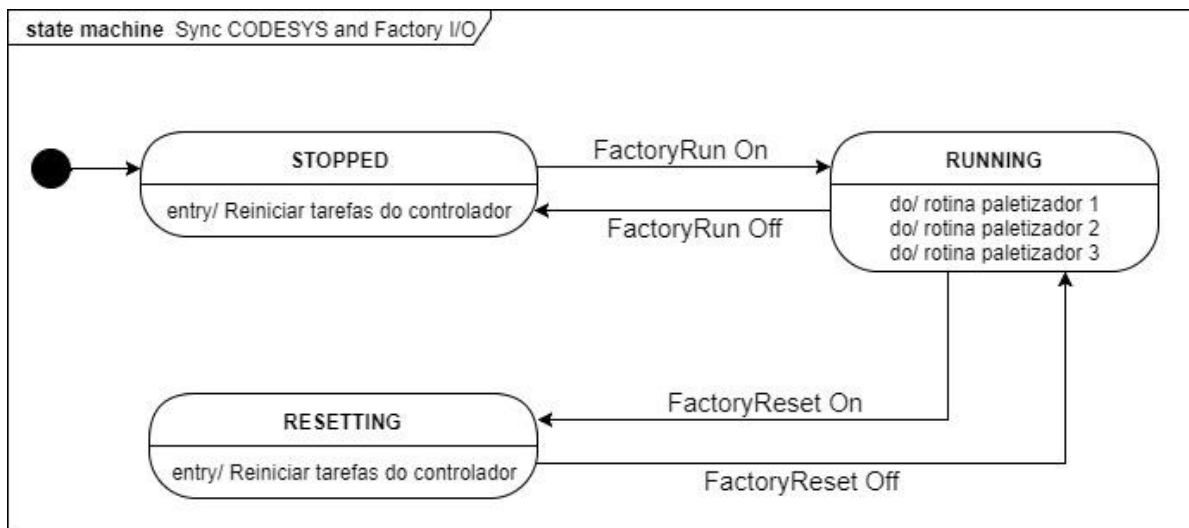


Figura A.1.6 – Diagrama de estados para sincronização do *SoftPLC CODESYS Win V3* com o *Factory I/O*.

A.2 CONFIGURAÇÃO DA COMUNICAÇÃO POR “DATA SOURCE”

A figura A.2.1 que se segue exemplifica a configuração do objeto “*Data Source*” no *Codesys*. Optou-se pelo tipo de comunicação simbólica.

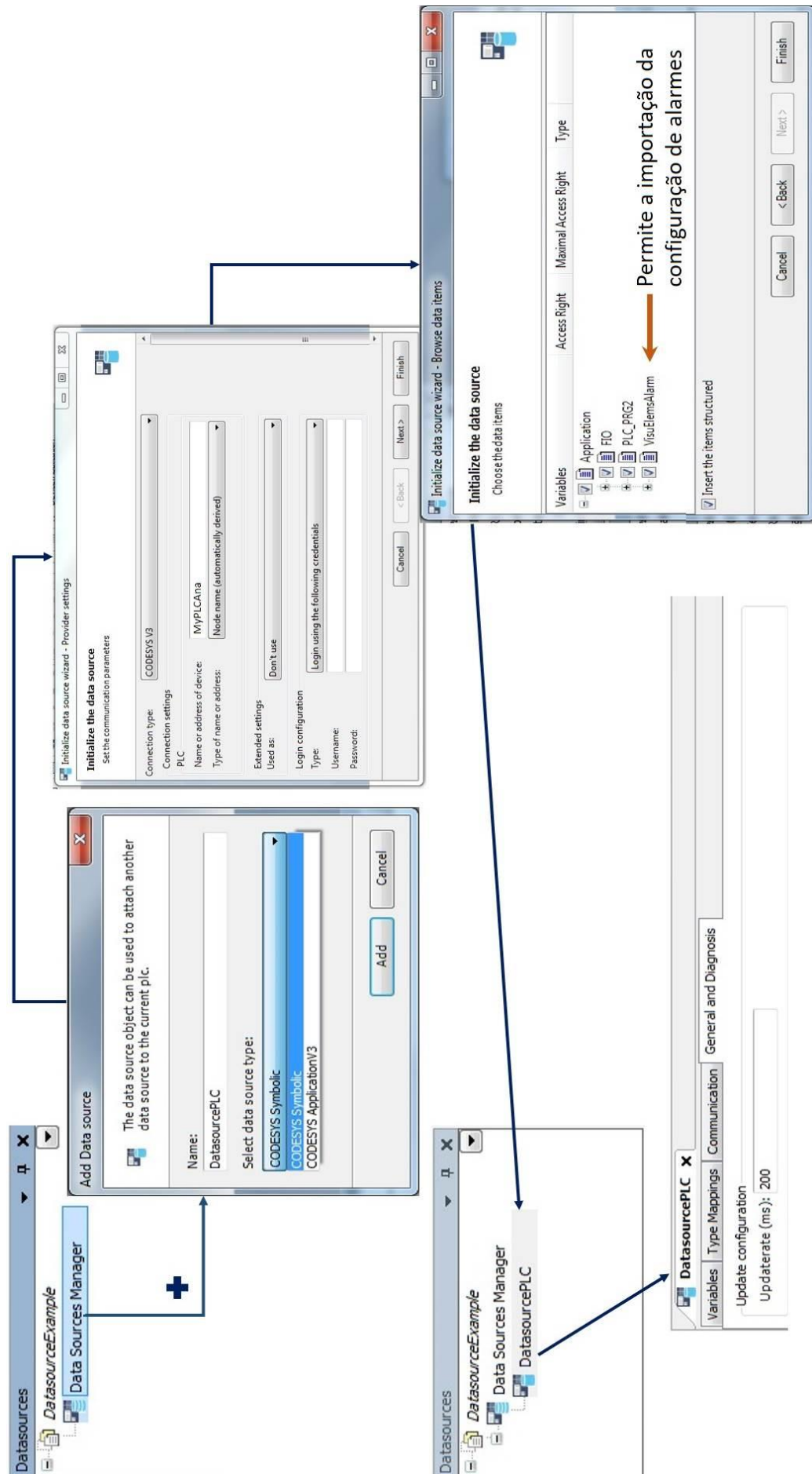


Figura A.2.1 – Configuração Data Source para as interfaces de manutenção e de supervisão.

B ANEXO: DETALHES DO DESENVOLVIMENTO DE SOLUÇÕES EM WinCC

Este anexo tem como objetivo explicar de forma mais detalhada o desenvolvimento das soluções de visualização desenvolvidas em WinCC.

A secção que se segue expõe os modelos de controlo utilizados na aplicação de encaminhamento, disponível na figura 5.3.

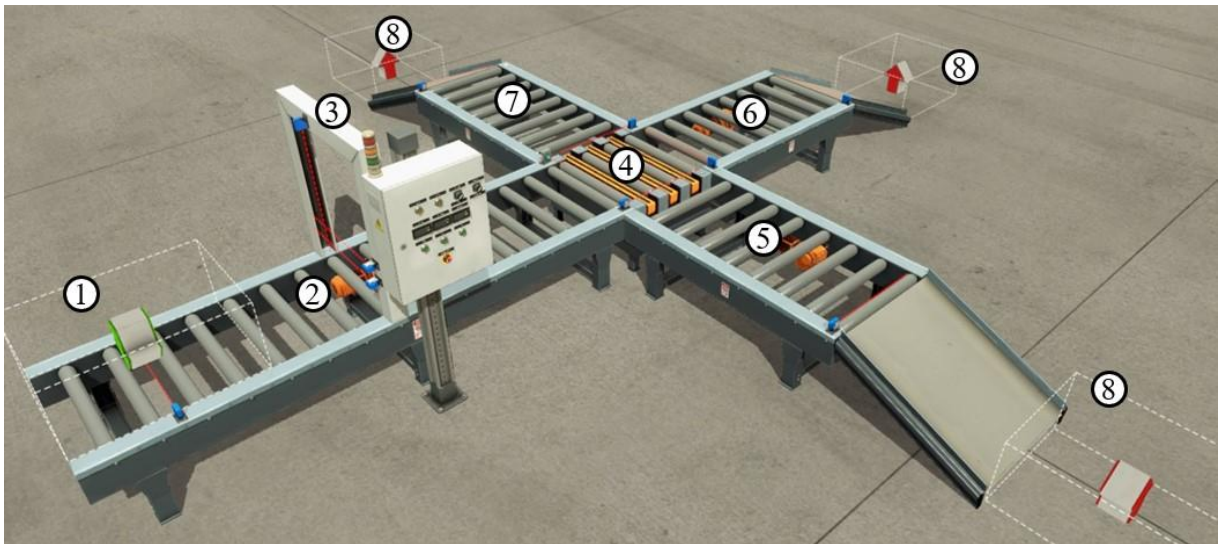


Figura 5.3 – Encaminhador de caixas, consoante a sua dimensão, desenvolvido no Factory I/O.

B.1 MODELOS DE CONTROLO DO ENCAMINHADOR

Para modelar o controlo do encaminhador de caixas deste caso de estudo utilizou-se uma abordagem semelhante à utilizada no caso anterior.

A figura B.1.1 apresenta o Grafcet G0 de alto nível, que mapeia as diferentes etapas de operação do encaminhador, sendo cada uma delas descritas em mais detalhe na tabela B.1.1. Dado tratar-se de uma aplicação mais simples do que no caso do paletizador, não se justificava a utilização das várias etapas de paragem anteriores, pelo que se adotaram apenas três.

Tabela B.1.1 – Descrição das diferentes etapas do Grafcet comportamental de alto nível dos modos de operação e de paragem do encaminhador.

Estados Grafcet G0	Descrição
A1	Estado de paragem ou inatividade (<i>Standby</i>), saídas desativadas.
D1	Estado de paragem de emergência, congela Grafkets de baixo nível nos estados atuais e desativa as saídas.
D2	Estado de paragem por falha no equipamento, inibe transição de estados nos Grafkets de baixo nível e desativa as saídas.
F1	Estado de corrida em modo de operação automático.
F2	Estado de corrida em modo de operação semiautomático.
F3	Estado de corrida de purga do equipamento, esvazia a máquina.
F4	Estado de corrida em modo de operação manual.

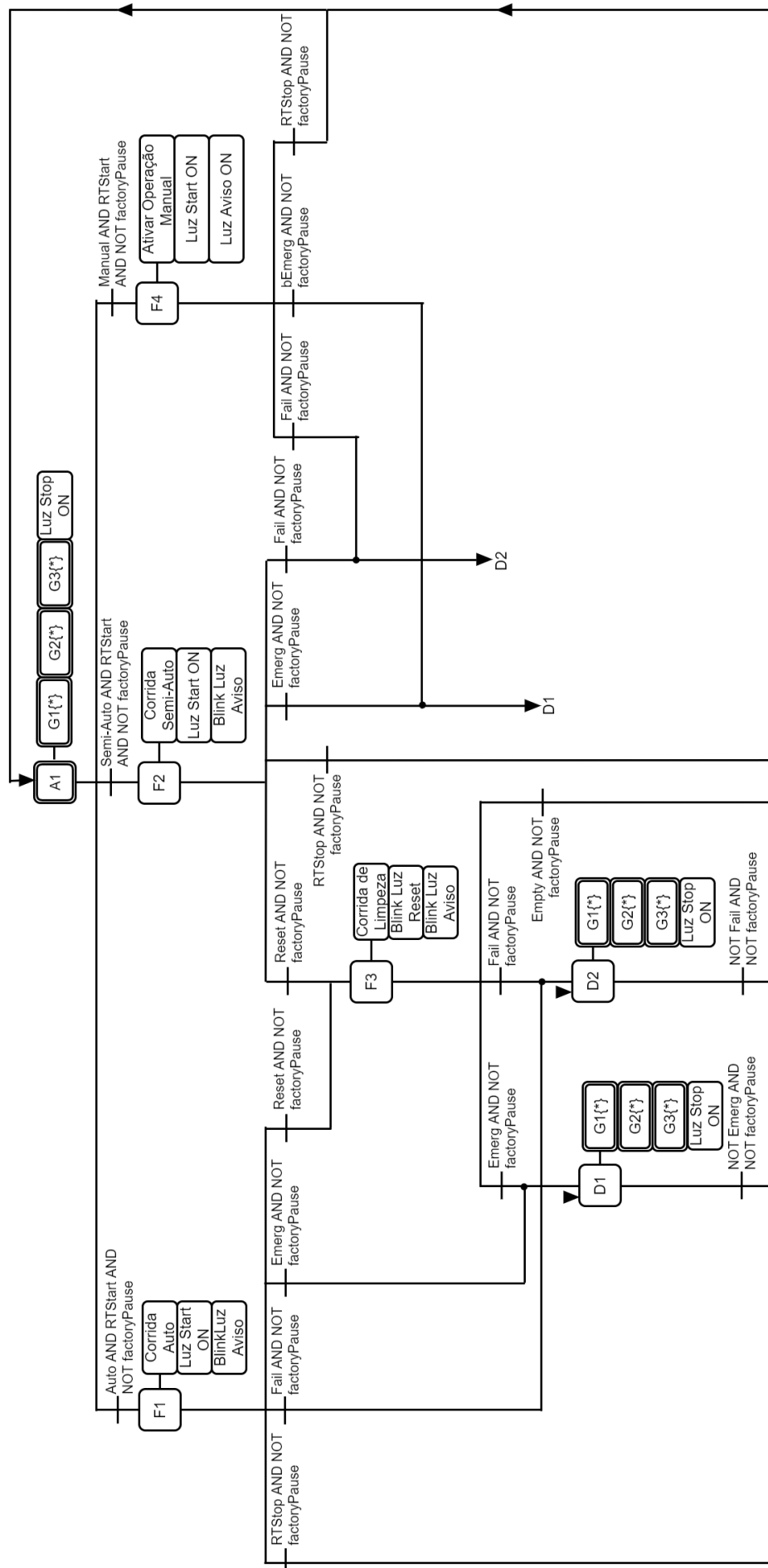


Figura B.1.1 – Grafset comportamental G0 de alto nível dos modos de operação e de paragem do encaminhador.

Por sua vez, subdividiu-se o encaminhador em cinco elementos alvo de controlo direto: o tapete de rolos de entrada, a mesa de transferência e os três tapetes de rolos de saída. Uma vez que os tapetes de saída apresentam um comportamento igual, foi apenas necessário descrever três Graficets para controlo direto:

- G1 para o controlo direto do tapete de entrada;
- G2 para o controlo direto da mesa de transferência;
- G3 para o controlo direto do tapete de saída, a replicar três vezes.

As figuras B.1.2, B.1.3 e B.1.4 apresentam os Grafkets G1, G2 e G3, respectivamente. É de notar que o sensor de saída do tapete de entrada e os sensores entrada dos tapetes de saída constituem os sensores de entrada e de saída da mesa de transferência, respectivamente.

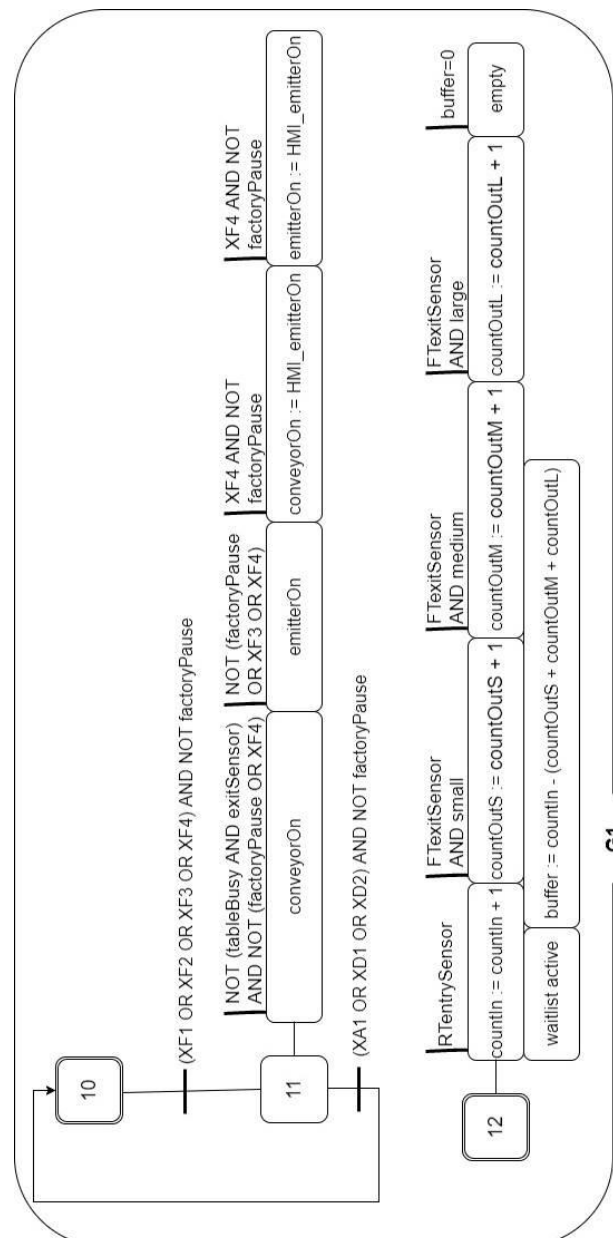


Figura B.1.2 – Grafset comportamental G1 do tapete de rolos de entrada.



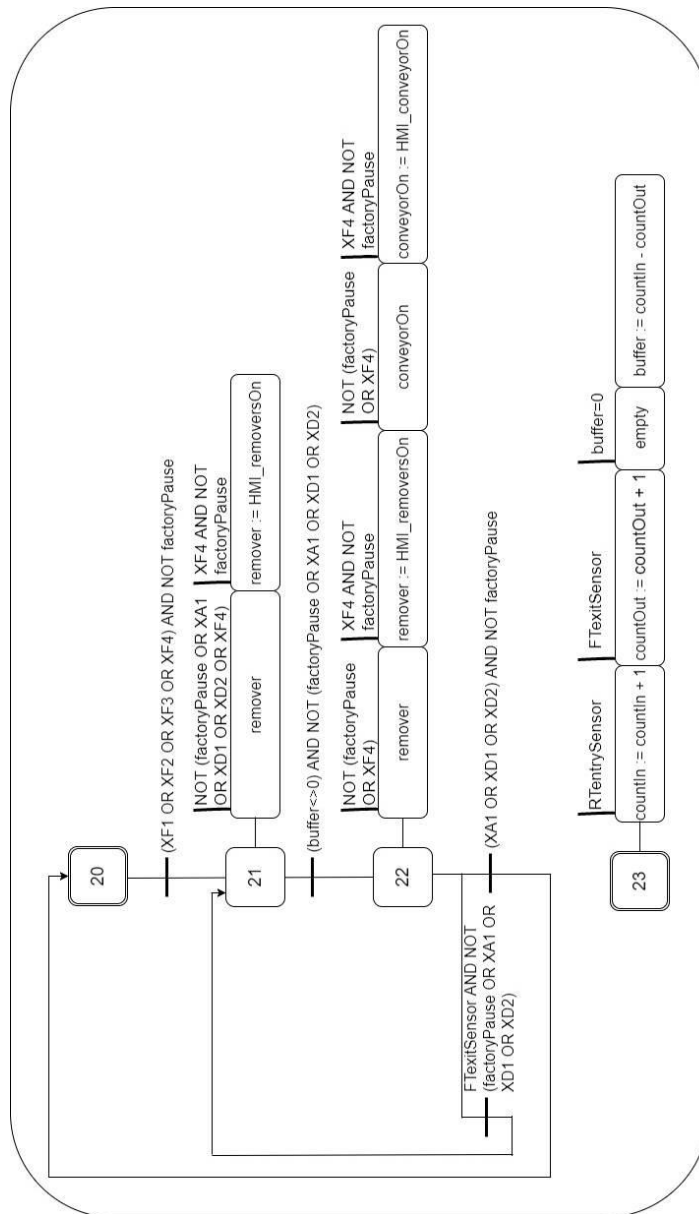


Figura B.1.4 – Grafcet comportamental G3 de um tapete de rolos de saída.

Note-se ainda que nos Grafcets G1, G2 e G3 surgem variáveis X^* , que representam as etapas de atividade do Grafcet G0, e, desta forma, demonstram a influência hierárquica de G0 sobre G1, G2 e G3. Para além disto, em todos os Grafcets é adotada a notação RT^* e FT^* para indicar a deteção da variação do valor de uma variável booleana de 0 para 1 e de 1 para 0, respetivamente.

Por fim, tal como no anexo anterior, utilizou-se novamente a máquina de estados da figura A.1.6, de modo a garantir sincronismo entre o cenário virtual e o PLC. Assim, sempre que o cenário virtual é suspenso ou reiniciado, é feita uma reinicialização às variáveis no PLC.